

**This Page Is Inserted by IFW Operations
and is not a part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- **BLACK BORDERS**
- **TEXT CUT OFF AT TOP, BOTTOM OR SIDES**
- **FADED TEXT**
- **ILLEGIBLE TEXT**
- **SKEWED/SLANTED IMAGES**
- **COLORED PHOTOS**
- **BLACK OR VERY BLACK AND WHITE DARK PHOTOS**
- **GRAY SCALE DOCUMENTS**

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

IN THE U.S. PATENT AND TRADEMARK OFFICE

#3
4-302^{ad}
jc927 U.S. PRO
10/021020
12/19/01

Applicant(s): USHIRO, Tadahiro

Application No.:

Group:

Filed: December 19, 2001

Examiner:

For: MICROCOMPUTER

L E T T E R

Assistant Commissioner for Patents
Box Patent Application
Washington, D.C. 20231

December 19, 2001
2936-0144P

Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55(a), the applicant hereby claims the right of priority based on the following application(s):

<u>Country</u>	<u>Application No.</u>	<u>Filed</u>
JAPAN	2000-395170	12/26/00

A certified copy of the above-noted application(s) is(are) attached hereto.

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to deposit Account No. 02-2448 for any additional fees required under 37 C.F.R. 1.16 or under 37 C.F.R. 1.17; particularly, extension of time fees.

Respectfully submitted,

BIRCH, STEWART, KOLASCH & BIRCH, LLP

By Clarence J. Birch #2921
TERRELL C. BIRCH

Attachment
(703) 205-8000
/ka

Reg. No. 19,382
P. O. Box 747
Falls Church, Virginia 22040-0747

日本国特許庁
JAPAN PATENT OFFICE

SHIRO, Tadahiro
Dec. 19, 2001
BSKB, LLP
(703) 205-8000
2936-0144P
10F1

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日
Date of Application:

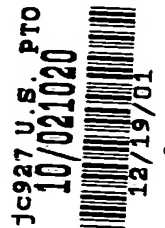
2000年12月26日

出願番号
Application Number:

特願2000-395170

出願人
Applicant(s):

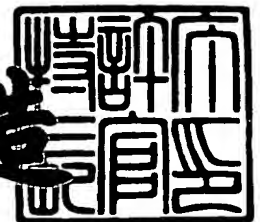
シャープ株式会社



2001年 9月11日

特許庁長官
Commissioner,
Japan Patent Office

及川耕造



出証番号 出証特2001-3083849

【書類名】 特許願

【整理番号】 00J04323

【提出日】 平成12年12月26日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 13/00

【発明の名称】 マイクロコンピュータ

【請求項の数】 7

【発明者】

【住所又は居所】 大阪府大阪市阿倍野区長池町 2 2 番 2 2 号 シャープ株式会社内

【氏名】 後呂 忠広

【特許出願人】

【識別番号】 000005049

【氏名又は名称】 シャープ株式会社

【代理人】

【識別番号】 100085501

【弁理士】

【氏名又は名称】 佐野 静夫

【手数料の表示】

【予納台帳番号】 024969

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9003086

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マイクロコンピュータ

【特許請求の範囲】

【請求項 1】 複数の周辺回路を備えたマイクロコンピュータにおいて、複数の周辺回路間の接続関係をプログラムで制御するための接続回路を備えたことを特徴とするマイクロコンピュータ。

【請求項 2】 前記接続回路は、プログラムによって与えられるデータに応じて複数の入力の中からいずれかを選択して出力するセレクトから成ることを特徴とする請求項 1 に記載のマイクロコンピュータ。

【請求項 3】 周辺回路に入力されるデータをプログラムで書き込むための回路を周辺回路として備えたことを特徴とする請求項 1 に記載のマイクロコンピュータ。

【請求項 4】 周辺回路から出力されるデータをプログラムで読み出すための回路を周辺回路として備えたことを特徴とする請求項 1 に記載のマイクロコンピュータ。

【請求項 5】 外部からの信号を内部に入力するための回路を周辺回路として備えたことを特徴とする請求項 1 に記載のマイクロコンピュータ。

【請求項 6】 内部で発生した信号を外部に出力するための回路を周辺回路として備えたことを特徴とする請求項 1 に記載のマイクロコンピュータ。

【請求項 7】 前記複数の周辺回路間の接続関係を前記接続回路を用いて制御することによって、所定の周辺回路機能を実現する手段を備えたことを特徴とする請求項 1 ～ 6 のいずれか 1 つに記載のマイクロコンピュータ。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、複数の周辺回路を備えたマイクロコンピュータに関するものである。

【 0 0 0 2 】

【従来の技術】

マイクロコンピュータは、中央処理装置（CPU）、データ及びプログラム用のメモリの他にI/Oポート、タイマカウンタなどの各種の周辺回路で構成されている。また、マイクロコンピュータには専用と汎用とがあり、専用マイクロコンピュータはASICとも呼ばれている。

【0003】

専用マイクロコンピュータの場合、個別の用途に応じて周辺回路を設計するため、効率の良い処理能力を期待できるが、その反面、新規設計部分の開発期間が必要であり、マイクロコンピュータを組み込む装置の開発日程によっては、開発が間に合わないことがある。

【0004】

汎用マイクロコンピュータは、様々な用途への対応を考慮するため、その機能は多様化している。しかしながら、それらの機能または性能が対象にするアプリケーションが要求する仕様を満たすとは限らない。そのため、要求される仕様を満足することができるような機種を選択することになるが、その機種に搭載されている機能が要求される仕様を満足することができるか否かの見極めが難しく、その見極めを誤った場合、プログラム設計中にマイクロコンピュータの機種変更やアプリケーションの仕様変更を招くという事態にもなりかねない。いずれにせよ、マイクロコンピュータの機種選択を誤った場合、アプリケーションの開発が遅れてしまう。

【0005】

汎用マイクロコンピュータでは、複数の周辺回路を連動させて動作させることにより、高機能を実現しようと工夫されている。例えば、タイマカウンタとI/Oポートとの連動により実現する「インプットキャプチャ」という機能がある。

【0006】

この機能は、外部入力信号の立ち上がりまたは立ち下がりエッジが検出される度にタイマのカウント値を専用のレジスタに格納するというものであり、外部入力信号がハイレベルまたはローレベルである時間を測定する場合などに有効である。

【0007】

タイマが 4 [MHz] のシステムクロックで動作しており、外部入力信号としてハイレベルが 1.5 [μ s]、ローレベルが 0.5 [μ s] のパルスが入力される場合を例に挙げて、外部入力信号の時間測定について図 2 1 を用いて説明する。外部入力信号 EXT__IN が立ち上がるまたは立ち下がる毎に、タイマのカウント値 T が専用のレジスタ R に格納され、その後、カウント値 T がリセットされてからタイマがカウントを再開する。

【 0 0 0 8 】

また、外部入力信号 EXT__IN が立ち上がるまたは立ち下がる毎に、割り込み要求が発生する。外部入力信号 EXT__IN による割り込み要求発生時のソフト処理では、そのフローチャートを図 2 2 に示すように、割り込み要求が発生した原因が外部入力信号 EXT__IN の立ち上がりであるか否かを判定する (S 8 0 1)。立ち上がりであれば (S 8 0 1 の Yes)、レジスタ R の値を外部入力信号 EXT__IN のローレベルである時間 (以下、ロー時間) として待避させる (S 8 0 2)。一方、立ち上がりでなければ (S 8 0 1 の No)、レジスタ R の値を外部入力信号 EXT__IN のハイレベルである時間 (以下、ハイ時間) として待避させる (S 8 0 3)。

【 0 0 0 9 】

ここで、タイマのカウント値は同じレジスタに上書きされるので、新しい値が上書きされる前にレジスタの値を待避させる必要があるが、今の例では、外部入力信号のハイ時間を待避するのに使える時間は 0.5 [μ s] しかない。これは 4 [MHz] のシステムクロックで動作する CPU では 2 ステートに相当するが、2 ステート以内に割り込み処理のある番地までジャンプして、割り込みソフト処理を実行し終えるのが不可能である場合が考えられる。

【 0 0 1 0 】

この場合、測定を可能にするためには、2 つのインプットキャプチャ機能を使用する必要がある。すなわち、インプットキャプチャ機能をもつ 2 つの I/O ポートに測定対象の信号を入力し、2 つのタイマで別々の時間を測定する。これについて図 2 3 を用いて説明する。

【 0 0 1 1 】

一方のインプットキャプチャ機能により、その外部入力信号EXT_IN1の立ち上がり及び立ち下がり、タイマのカウント値T1がレジスタR1に格納された後、カウント値T1がリセットされてからタイマがカウントを再開する。そして、他方のインプットキャプチャ機能により、その外部入力信号EXT_IN2が立ち下がる毎に、タイマのカウント値T2がレジスタR2に格納された後、カウント値T2がリセットされてからタイマがカウントを再開する。

【0012】

また、外部入力信号EXT_IN1の立ち上がり及び立ち下がり、並びに、外部入力信号EXT_IN2の立ち下がり、で割り込み要求が発生する。割り込み要求発生時のソフト処理のフローチャートを図24及び図25に示す。外部入力信号EXT_IN1により割り込み要求が発生した場合、割り込み要求が発生した原因が外部入力信号EXT_IN1の立ち上がりであれば(S901のYes)、対応するレジスタR1の値をロー時間として待避させる(S902)。一方、外部入力信号EXT_IN2により割り込み要求が発生した場合、対応するレジスタR2の値からレジスタR1に待避させているロー時間を引いた値をハイ時間として待避させる(S1001)。

【0013】

しかしながら、この方法では、タイマ以外にI/Oポートを余分に使ってしまうことになる。マイクロコンピュータが専用チップであれば、単一の端子からの信号を異なる回路ブロックに供給することになる。さらに、問題となるのは、この方法にプログラマが思い至るまでに消費する時間である。

【0014】

また、プログラマがこの方法を考えついたとしても、インプットキャプチャ機能をもつI/Oポートが余っている必要がある。マイクロコンピュータを組み込む装置が開発される経緯で、マイクロコンピュータのプログラム開発が行われているときには、マイクロコンピュータを含む回路は決定している場合がほとんどである。

【0015】

このようなことから、マイクロコンピュータのプログラミング中に特殊な機能

をもつ I/O ポートの仕様割り当てを変更しなければならない事態が発生すれば、装置全体の開発計画に影響を及ぼしかねない。また、装置設計の初期に、すなわち、マイクロコンピュータの機種選択時に、このような特殊な機能をもつ I/O ポートの 2 つもが 1 つの信号測定のために占有されるという事態を予想できるかどうか疑問である。

【 0 0 1 6 】

このように、汎用マイクロコンピュータを特殊なオペレーションを含んだ装置に組み込む場合、予想外の事態が発生することは少なくなく、それが装置全体の開発計画の遅延を招くこともしばしば起こる。

【 0 0 1 7 】

尚、専用マイクロコンピュータの開発時間がとれず、汎用コンピュータの周辺回路機能で装置の供給仕様を満足することができるか否かの見極めが難しい場合、特開平 5 - 1 2 7 9 1 3 号の公報に開示されているように、プログラマブルゲートアレイで所望の周辺回路をプログラムと同時に開発する方法も考えられる。

【 0 0 1 8 】

しかしながら、プログラムゲートアレイの原理からして、同じ周辺回路機能を実現するためには回路面積が大きくなる。また、必要な周辺回路機能が不明なため、ゲート数を多めに確保しておく必要があり、専用マイクロコンピュータよりもコストが高くなる。さらに、プログラマブルゲートアレイの製造プロセスは特殊であるため、これらを同一チップ上に製造するためには特殊な技術が必要になり、装置に組み込むためのマイクロコンピュータで実用になっていないのが現状である。

【 0 0 1 9 】

【発明が解決しようとする課題】

このように、専用コンピュータでは開発日程が長いという問題があり、汎用マイクロコンピュータでは装備されている周辺回路機能が組み込み対象の装置が要求する仕様を満足するか否かの見極めが難しく、プログラム開発中に装置の要求仕様を満足しないことが判明した場合、汎用マイクロコンピュータの機種変更や装置の仕様変更を余儀なくされる。

【 0 0 2 0 】

汎用マイクロコンピュータの限定された周辺回路機能で装置の要求仕様を満たす方法をプログラマが考え出したとしても、それがプログラムだけで対応可能とは限らず、装置全体の回路変更が必要な場合もある。また、そのような特殊な解決方法をプログラマが考え出すのにも時間を要する。いずれにせよ、装置全体の開発計画の遅延につながる。

【 0 0 2 1 】

また、装置設計においては製造コストの問題から選択できる汎用マイクロコンピュータの機種が限定されることがほとんどであるにもかかわらず、装置の要求仕様を満足する周辺回路機能がコスト的に上の機種にしか用意されていない場合などには、選択の余地が与えられず、コスト高を招く。

【 0 0 2 2 】

そこで、本発明は、特殊な製造技術や製造工程を使用することなく、また、専用マイクロコンピュータのように長期の開発日程を要することなく、汎用マイクロコンピュータよりも専用マイクロコンピュータに近い高機能な周辺回路機能を実現することができるようにしたマイクロコンピュータを提供することを目的とする。

【 0 0 2 3 】

【課題を解決するための手段】

上記の目的を達成するため、本発明では、複数の周辺回路を備えたマイクロコンピュータにおいて、複数の周辺回路間の接続関係をプログラムで制御するための接続回路を備えている。

【 0 0 2 4 】

この構成により、個々の基本的な周辺回路を、接続のための回路を用いてプログラムで接続することによって、個々の基本的な周辺回路では実現し得ない高機能を実現することができるようになる。

【 0 0 2 5 】

【発明の実施の形態】

以下に、本発明の実施形態を図面を参照しながら説明する。本発明の一実施形

態であるマイクロコンピュータのブロック図を図1に示す。同図において、1はCPU、2は入力レジスタ、3は出力レジスタ、4は接続回路、5は第1のタイマ、6は第2のタイマ、7は論理回路、8はデータバスである。

【0026】

周辺回路の1つである第1のタイマ5、第2のタイマ6はそれぞれ図2、図3に示すようにダウンカウンタで構成されている。ダウンカウンタの動作は次の通りである。

【0027】

カウント値は不図示のクロック信号に同期して1ずつ小さくなる。端子Startへの入力の立ち上がりエッジでカウントを開始する。端子Stopへの入力の立ち上がりエッジでカウントを停止する。端子OUT0～OUT7からは現在のカウント値を示す8ビットの信号を出力する。カウント値にオーバーフローが発生したとき（カウント値が0x00になったとき）には、端子Overflowからの出力がハイレベルになる。

【0028】

端子Resetへの入力の立ち上がりエッジでリセットがかかる。すなわち、端子IN0～IN7に入力される8ビットの信号が示す値にカウント値が設定されるとともに、端子Overflowからの出力がローレベルになる。また、端子Resetへの入力が高レベルである間は、端子Startへの入力は無効になる（端子Startへの入力に立ち上がりエッジが生じてもカウントを開始することはない）。

【0029】

尚、本実施形態では、第1のタイマ5にオーバーフローが発生すると、割り込み要求を発生する回路、及び、第2のタイマ6にオーバーフローが発生すると、割り込み要求を発生する回路が設けられている。

【0030】

周辺回路の基本部品の1つである論理回路7は、図4に示すように、3入力のORゲート701、フリップフロップ702、インバータ703、ANDゲート704、ANDゲート705、ORゲート706、及び、インバータ707から

成っている。

【 0 0 3 1 】

入力信号 IN 1、IN 2、IN 3 は OR ゲート 7 0 1 に入力されている。フリップフロップ 7 0 2 では、データ端子 D に自身の反転端子 Q' から出力される信号が入力されており、クロック端子 CK に OR ゲート 7 0 1 から出力される信号が入力されており、リセット端子 R に入力信号 D _ R E S E T が入力されている。インバータ 7 0 3 には入力信号 T R G が入力されている。

【 0 0 3 2 】

AND ゲート 7 0 4 には、OR ゲート 7 0 1 から出力される信号、及び、インバータ 7 0 3 から出力される信号が入力されている。AND ゲート 7 0 5 には、入力信号 T R G、及び、フリップフロップ 7 0 2 の反転端子 Q' から出力される信号が入力されている。

【 0 0 3 3 】

OR ゲート 7 0 6 には AND ゲート 7 0 4 から出力される信号、及び、AND ゲート 7 0 5 から出力される信号が入力されている。OR ゲート 7 0 6 から出力される信号は出力信号 O U T 1 として出力されるとともに、インバータ 7 0 7 を介して出力信号 O U T 2 として出力される。

【 0 0 3 4 】

以上より、論理回路 7 では、入力信号 T R G がローレベルであるときには、出力信号 O U T 1 及び O U T 2 には OR ゲート 7 0 1 から出力される信号が反映され、入力信号 IN 1、IN 2、IN 3 のいずれか 1 つでもハイレベルならば、出力信号 O U T 1 はハイレベル、出力信号 O U T 2 はローレベルとなり、入力信号 IN 1、IN 2、IN 3 の全てがローレベルならば、出力信号 O U T 1 はローレベル、出力信号 O U T 2 はハイレベルとなる。

【 0 0 3 5 】

一方、入力信号 T R G がハイレベルであるときには、入力信号 D _ R E S E T がハイレベルであれば、出力信号 O U T 1 はハイレベル、出力信号 O U T 2 はローレベルであり、また、入力信号 D _ R E S E T がローレベルであるときには、入力信号 IN 1、IN 2、IN 3 について、全てがローレベルである状態から、

いずれか1つでもハイレベルである状態になる毎に、出力信号OUT1及びOUT2が反転する。

【0036】

周辺回路の1つである入力レジスタ2は、図5に示すように32ビットのラッチ回路で構成されており、端子D0～D31に入力される32ビットの信号INREG0～INREG31の値を不図示のクロック信号に同期して取り込むとともに、CPU1が予め決められた特定のアドレスをリードしようとするときにアドレスデコーダから出力される信号CPU_RDがハイレベルであるときには、信号INREG0～INREG31の最後に取り込んだ値を端子O0～O31からデータバス8上に出力する。すなわち、信号CPU_RDがハイレベルであるときには、データバス8上の32ビットの信号CPU_BUS0～31の値は、入力レジスタ2によって最後に取り込まれた信号INREG0～INREG31の値になる。

【0037】

入力レジスタ2にはCPU1のメモリ領域としてアドレス0xF0000000が割り当てられており、この割り当てられたアドレスをCPU1のプログラムでリードアクセスすると、信号CPU_RDがハイレベルになるようになっている。

【0038】

周辺回路の1つである出力レジスタ3は、図6に示すように32ビットのラッチ回路で構成されており、CPU1が予め決められた特定のアドレスにライトしようとするときにアドレスデコーダから出力される信号CPU_WRがハイレベルであるときには、不図示のクロック信号に同期してデータバス8上の32ビットの信号CPU_BUS0～31の値を端子D0～D31から取り込むとともに、最後に取り込んだ値を端子O0～O31から出力する。出力レジスタ3の端子O0～O31から出力される32ビットの信号OUTREG0～31は接続回路4に入力されている。

【0039】

尚、出力レジスタ3にはCPU1のメモリ領域としてアドレス0xF0000004が割り当てられており、この割り当てられたアドレスをCPU1のプログラムでライ

トアクセスすると、信号CPU_WRがハイレベルになるようになっている。また、信号CPU_WRがローレベルであるときには、出力レジスタ3の端子O0～O31から出力される信号OUTREG0～OUTREG31は変化しないようになっている。

【0040】

接続回路4は、図7に示すセクタ401～412、及び、図8に示すセクタ413～416で構成されている。セクタ401～412は、端子S0、S1、及び、S2に入力される信号の状態に応じて、端子D0、D1、D2、D3、D4、D5の中から1つを選択し、選択した端子に入力される信号を端子OUTから出力する。端子S0、S1、及び、S2に入力される信号の状態と、選択される端子との関係を図9に示しておく。尚、同図中において、1はハイレベルであること、0はローレベルであることをそれぞれ示す。

【0041】

但し、端子CSに入力される信号がローレベルであるときには、端子S0、S1、及び、S2に入力される信号の状態が変化しても、端子D0、D1、D2、D3、D4、D5の選択状態は変化しない、言い換えれば、端子OUTから出力される信号は変化しない。

【0042】

セクタ401～412の端子D0、D1、D2、D3、D5には、それぞれ第1のタイマ5の端子Overflowから出力される信号OVERFLOW1、第2のタイマ6の端子Overflowから出力される信号OVERFLOW2、論理回路7の出力信号OUT1、OUT2、マイクロコンピュータの外部から入力される割り込み付き外部入力信号EXT_IN/INTが入力されている。尚、割り込み付き外部入力信号とは、その立ち上がりエッジ及び立ち下がりエッジで割り込み要求が発生する回路に入力されている信号のことである。

【0043】

セクタ401、402、…、412の端子D4には、それぞれ出力レジスタ3の端子O16から出力される信号OUTREG16、端子O17から出力される信号OUTREG17、…、端子O27から出力される信号OUTREG27

が入力されている。

【0044】

セクタ401の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS0、CPU__BUS1、CPU__BUS2が入力されている。セクタ402の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS3、CPU__BUS4、CPU__BUS5が入力されている。セクタ403の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS6、CPU__BUS7、CPU__BUS8が入力されている。セクタ404の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS9、CPU__BUS10、CPU__BUS11が入力されている。セクタ405の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS12、CPU__BUS13、CPU__BUS14が入力されている。セクタ406の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS15、CPU__BUS16、CPU__BUS17が入力されている。

【0045】

セクタ407の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS18、CPU__BUS19、CPU__BUS20が入力されている。セクタ408の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS21、CPU__BUS22、CPU__BUS23が入力されている。セクタ409の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS24、CPU__BUS25、CPU__BUS26が入力されている。セクタ410の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS27、CPU__BUS28、CPU__BUS29が入力されている。セクタ411の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS0、CPU__BUS1、CPU__BUS2が入力されている。セクタ412の端子S0、S1、S2には、それぞれデータバス8上の信号CPU__BUS3、CPU__BUS4、CPU__BUS5が入力されている。

【0046】

セクタ401～410の各端子CSには、CPU1のアドレスデコーダから出力される信号CPU_WR_S1が入力されている。セクタ411及び412の各端子CSには、CPU1のアドレスデコーダから出力される信号CPU_WR_S2が入力されている。

【0047】

尚、セクタ401～410には、CPU1のメモリ領域としてアドレス0xF000008が割り当てられており、この割り当てられたアドレスをCPU1のプログラムでライトアクセスすると、信号CPU_WR_S1がハイレベルになるようになっている。また、セクタ411及び412には、CPU1のメモリ領域としてアドレス0xF00000Cが割り当てられており、この割り当てられたアドレスをCPU1のプログラムでライトアクセスすると、信号CPU_WR_S2がハイレベルになるようになっている。

【0048】

セクタ401の端子OUTから出力される信号は、第1のタイマ5の端子Startに入力される信号START1になっている。セクタ402の端子OUTから出力される信号は、第1のタイマ5の端子Resetに入力される信号RESET1になっている。セクタ403の端子OUTから出力される信号は、第1のタイマ5の端子Stopに入力される信号STOP1になっている。

【0049】

セクタ404の端子OUTから出力される信号は、第2のタイマ6の端子Startに入力される信号START2になっている。セクタ405の端子OUTから出力される信号は、第2のタイマ6の端子Resetに入力される信号RESET2になっている。セクタ406の端子OUTから出力される信号は、第2のタイマ6の端子Stopに入力される信号STOP2になっている。

【0050】

セクタ407、408、409、410、411の端子OUTから出力される信号は、それぞれ論理回路7の入力信号IN1、IN2、IN3、TRG、D_RESETになっている。セクタ412の端子OUTから出力される信号はマイクロコンピュータの外部に出力される信号EXT_OUTになっている。

【0051】

セクタ413～416は、端子S0に入力される信号の状態に応じて、端子D1__0、D1__1、…、D1__7の組み合わせと、端子D2__0、D2__1、…、D2__7の組み合わせとのいずれか一方を選択し、選択した組み合わせの端子に入力される8ビットの信号を端子O0～O7から出力する。端子S0に入力される信号がローレベルであるときには、端子D1__0、D1__1、…、D1__7の組み合わせを選択し、一方、端子S0に入力される信号がハイレベルであるときには、端子D2__0、D2__1、…、D2__7の組み合わせを選択する。尚、端子CSに入力される信号がローレベルであるときには、端子S0に入力される信号の状態が変化しても、端子の選択状態は変化しない、言い換えれば、端子O0～O7から出力される信号は変化しない。

【0052】

セクタ413及び414の端子D1__0、D1__1、…、D1__7、及び、D2__0、D2__1、…、D2__7には、それぞれ出力レジスタ3から出力される下位16ビットの信号OUTREG0、1、…、7、8、9、…、15が入力されている。

【0053】

セクタ415及び416では、端子D1__0～D1__7には第1のタイマ5の端子OUT0～OUT7から出力される8ビットの信号DOUT1__0～DOUT1__7が、端子D2__0～D2__7には第2のタイマ6の端子OUT0～OUT7から出力される8ビットの信号DOUT2__0～DOUT2__7がそれぞれ入力されている。

【0054】

セクタ413、414、415、416の端子S0には、それぞれデータバス8上の信号CPU__BUS0、CPU__BUS1、CPU__BUS2、CPU__BUS3が入力されている。

【0055】

セクタ413～416の各端子CSには、CPU1のアドレスデコーダから出力される信号CPU__WR__S3が入力されている。尚、セクタ413～4

16には、CPU1のメモリ領域としてアドレス0xF0000010が割り当てられており、この割り当てられたアドレスをCPU1のプログラムでライトアクセスすると、信号CPU__WR__S3がハイレベルになるようになっている。

【0056】

セクタ413の端子00～07から出力される信号は、第1のタイマ5の端子IN0～IN7に入力される8ビットの信号DIN1__0～DIN1__7になっている。セクタ414の端子00～07から出力される信号は、第2のタイマ6の端子IN0～IN7に入力される8ビットの信号DIN2__0～DIN2__7になっている。

【0057】

セクタ415の端子00～07から出力される信号は、入力レジスタ2の端子D0～D7に入力される信号INREG0～INREG7になっている。セクタ416の端子00～07から出力される信号は、入力レジスタ2の端子D8～D15に入力される信号INREG8～INREG15になっている。

【0058】

プログラムでは、データバス8への入出力には、不図示の32ビットのレジスタReg0～Reg31を使用する。すなわち、データバス8上の信号CPU__BUS0は、レジスタReg0のビットを1にするとハイレベルになり、レジスタReg0のビットを0にするとローレベルになる。また、データバス8上の信号CPU__BUS0がハイレベルであるときにレジスタReg0の値を評価すると1になり、信号CPU__BUS0がローレベルであるときにレジスタReg0の値を評価すると0になる。この関係は、レジスタReg1とデータバス8上の信号CPU__BUS1、レジスタReg2とデータバス8上の信号CPU__BUS2、…、レジスタReg31とデータバス8上の信号CPU__BUS31についても同じである。

【0059】

所定の機能をもった周辺回路を構築するためのプログラムのフローチャートの一例を図10に示す。まず、レジスタReg0=1、レジスタReg1=1、レジスタReg2=0、レジスタReg3=0、レジスタReg4=0、レジスタ

Reg 5 = 1、レジスタ Reg 6 = 0、レジスタ Reg 7 = 1、レジスタ Reg 8 = 0、レジスタ Reg 9 = 0、レジスタ Reg 10 = 1、レジスタ Reg 11 = 0、レジスタ Reg 12 = 0、レジスタ Reg 13 = 0、レジスタ Reg 14 = 1、レジスタ Reg 15 = 1、レジスタ Reg 16 = 1、レジスタ Reg 17 = 0、レジスタ Reg 18 = 1、レジスタ Reg 19 = 0、レジスタ Reg 20 = 1、レジスタ Reg 21 = 0、レジスタ Reg 22 = 0、レジスタ Reg 23 = 1、レジスタ Reg 24 = 0、レジスタ Reg 25 = 0、レジスタ Reg 26 = 1、レジスタ Reg 27 = 0、レジスタ Reg 28 = 0、レジスタ Reg 29 = 1にする (S101)。

【0060】

次に、アクセスするアドレスを接続回路4のセクタ401～410に設定する、すなわち、アドレス0xF0000008にライトアクセスする (S102)。これにより、セクタ401～410の端子CSに入力される信号CPU_WR_S1がハイレベルになり、セクタ401～410で選択される信号を切り換えられる。

【0061】

次に、レジスタ Reg 0 = 0、レジスタ Reg 1 = 0、レジスタ Reg 2 = 1にする (S103)。次に、アクセスするアドレスを接続回路4のセクタ411に設定する、すなわち、アドレス0xF000000Cにライトアクセスする (S104)。これにより、セクタ411の端子CSに入力される信号CPU_WR_S2がハイレベルになり、セクタ411で選択される信号が切り換えられる。

【0062】

S101～S104の処理により、論理回路7のインバータ707からの出力信号OUT2、出力レジスタ3の端子O17からの出力信号OUTREG17、論理回路7のANDゲート706からの出力信号OUT1が、それぞれ第1のタイマ5の端子Startへの入力信号START1、端子Resetへの入力信号RESET1、端子Stopへの入力信号STOP1になる。

【0063】

また、論理回路7のANDゲート706からの出力信号OUT1、出力レジス

タ3の端子O20からの出力信号OUTREG20、論理回路7のインバータ707からの出力信号OUT2が、それぞれ第2のタイマ6の端子Startへの入力信号START2、端子Resetへの入力信号RESET2、端子Stopへの入力信号STOP2になる。

【0064】

また、外部入力信号EXT__IN/INT、出力レジスタ3の端子O23からの出力信号OUTREG23、端子O24からの出力信号OUTREG24、端子O25からの出力信号OUTREG25、端子O26からの出力信号OUTREG26が、それぞれ論理回路7のORゲート701への入力信号IN1、IN2、IN3、インバータ703及びANDゲート705への入力信号TRG、フリップフロップ702の端子Rへの入力信号D__RESETになる。

【0065】

次に、レジスタReg0=0、レジスタReg1=1、レジスタReg2=0、レジスタReg3=1にする(S105)。次に、アクセスするアドレスを接続回路4のセクタ413~416に設定する、すなわち、アドレス0xF0000010にライトアクセスする(S106)。これにより、セクタ413~416の端子CSに入力される信号CPU__WR__S3がハイレベルになり、セクタ413~416で選択される信号が切り換えられる。

【0066】

S105及びS106の処理により、出力レジスタ3の端子O0~O7からの出力信号OUTREG0~OUTREG7が第1のタイマ5の端子IN0~IN7への入力信号DIN1__0~DIN1__7になり、出力レジスタ3の端子O8~O15からの出力信号OUTREG8~OUTREG15が第2のタイマ6の端子IN0~IN7への入力信号DIN2__0~DIN2__7になり、第1のタイマ5の端子OUT0~OUT7からの出力信号DOUT1__1~DOUT1__7が入力レジスタ2の端子D0~D7への入力信号INREG0~INREG7になり、第2のタイマ6の端子OUT0~OUT7からの出力信号DOUT2__1~DOUT2__7が入力レジスタ2の端子D8~D15への入力信号INREG8~INREG15になる。したがって、図11に示すような周辺回路が構築

される。

【0067】

次に、レジスタReg 0～7の値を全て1、レジスタReg 8～15の値を全て1、レジスタReg 17の値を1、レジスタReg 20の値を1、レジスタReg 23の値を0、レジスタReg 24の値を0、レジスタReg 25の値を0、レジスタReg 26の値を1にする（S107）。

【0068】

次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする（S108）。これにより、出力レジスタ3の入力信号CPU_WRがハイレベルになり、出力レジスタ3がデータバス8上の信号を取り込む状態になる。

【0069】

S107及びS108の処理により、第1のタイマ5及び第2のタイマ6では、端子IN0～IN7への入力が全てハイレベル、端子Resetへの入力がハイレベルになり、また、論理回路7では、ORゲート701への外部入力信号EXT_IN以外の2入力がローレベル、フリップフロップ702の端子Rへの入力がハイレベル、インバータ703への入力、及び、ANDゲート705への1入力がハイレベルになる。

【0070】

したがって、外部入力信号EXT_IN/INTが立ち上がる毎に、第1のタイマ5の端子Startへの入力、及び、第2のタイマ6の端子Stopへの入力が立ち上がる。また、外部入力信号EXT_IN/INTが立ち下がる毎に、第1のタイマ5の端子Stopへの入力、及び、第2のタイマ6の端子Startへの入力が立ち上がる。但し、第1のタイマ5の端子Resetへの入力、及び、第2のタイマ6の端子Resetへの入力がハイレベルであるので、第1のタイマ5及び第2のタイマ6にはリセットがかかった状態になっており、第1のタイマ5及び第2のタイマ6がカウント動作をすることはない。

【0071】

外部入力信号EXT_IN/INTのハイ時間及びロー時間の測定を開始する

ためのプログラムのフローチャートを図12に示す。まず、レジスタReg17の値を0、レジスタReg20の値を0、レジスタReg23の値を0、レジスタReg24の値を0、レジスタReg25の値を0、レジスタReg26の値を1にする(S201)。次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする(S202)。

【0072】

この処理により、論理回路7では、ORゲート701への外部入力信号EXT__IN以外の2入力がローレベル、フリップフロップ702の端子Rへの入力がハイレベル、インバータ703への入力、及び、ANDゲート705への1入力がハイレベルになる。また、第1のタイマ5及び第2のタイマ6の端子Resetへの入力がローレベルになり、第1のタイマ5及び第2のタイマ6のリセットが解除される。したがって、第1のタイマ5は、外部入力信号EXT__IN/INTの立ち下がりでカウント値0xFFからカウントを開始し、外部入力信号EXT__IN/INTの立ち上がりでカウントを停止する状態になる。また、第2のタイマ6は、外部入力信号EXT__IN/INTの立ち上がりでカウント値0xFFからカウントを開始し、外部入力信号EXT__IN/INTの立ち下がりでカウントを停止する状態になる。

【0073】

次に、外部入力信号EXT__IN/INTのハイ時間及びロー時間の測定を開始した後、外部入力信号EXT__IN/INTの立ち上がりあるいは立ち下がりに伴って割り込み要求が発生した場合に行うプログラムのフローチャートを図13に示す。まず、アクセスするアドレスを入力レジスタ2に設定する、すなわち、アドレス0xF0000000にリードアクセスする(S301)。次に、外部入力信号EXT__IN/INTの立ち上がりであるか否かを判定する(S302)。

【0074】

S302での判定の結果、立ち上がりであれば(S302のYes)、レジスタReg0~7の値をソフト処理で使用するRAMなどに待避させる(S303)。次に、レジスタReg0~7の値を全て1、レジスタReg17の値を1、レジスタReg20の値を0、レジスタReg23の値を0、レジスタReg2

4の値を0、レジスタReg 25の値を0、レジスタReg 26の値を1にする(S304)。次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする(S305)。次に、レジスタReg 17の値を0にする(S306)。次に、アクセスするアドレスを出力レジスタ3に設定する(S311)。

【0075】

一方、S302での判定の結果、立ち上がりでなければ(S302のNo)、レジスタReg 8～15の値をソフト処理で使用するRAMなどに待避させる(S307)。次に、レジスタReg 8～15の値を全て1、レジスタReg 17の値を0、レジスタReg 20の値を1、レジスタReg 23の値を0、レジスタReg 24の値を0、レジスタReg 25の値を0、レジスタReg 26の値を1にする(S308)。次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする(S309)。次に、レジスタReg 20の値を0にする(S310)。次に、アクセスするアドレスを出力レジスタ3に設定する(S311)。

【0076】

以上のプログラムにより、第1のタイマ5のカウント値T1及び第2のタイマ6のカウント値T2、並びに、レジスタReg 0～7及びレジスタReg 8～15の値が外部入力信号EXT_IN/INTに応じて図14に示すように変化する。但し、外部入力信号EXT_IN/INTのロー時間が0.5[μs]、ハイ時間が1.5[μs]、第1のタイマ5及び第2のタイマ6のクロック信号の周波数を4[MHz]であると想定している。

【0077】

すなわち、外部入力信号EXT_IN/INTが立ち上がると、第1のタイマ5のカウント値T1がレジスタReg 0～7に格納され、その後、レジスタReg 0～7の値が待避され、その後、第1のタイマ5に一時的にリセットがかかり、そのカウント値T1が0xFFに設定される。尚、外部入力信号EXT_IN/INTが立ち上がると、第1のタイマ5がカウントを停止するとともに、第2のタイマ6がカウント値0xFFからカウントを開始する。

【0078】

また、外部入力信号EXT_IN/INTが立ち下がると、第2のタイマ6のカウンタ値T2がレジスタReg8～15に格納され、その後、レジスタReg8～15の値が待避され、その後、第2のタイマ6には一時的にリセットがかかり、そのカウンタ値T2が0xFFに設定される。尚、外部入力信号EXT_IN/INTが立ち下がると、第2のタイマ6がカウンタを停止するとともに、第1のタイマ5がカウンタ値0xFFからカウンタを開始する。

【0079】

以上の動作において、外部入力信号EXT_IN/INTが立ち上がった時点で待避させた値がロー時間、外部入力信号EXT_IN/INTが立ち下がった時点で待避させた値がハイ時間になる。

【0080】

このように、本実施形態のマイクロコンピュータでは、個々の基本的な周辺回路（第1のタイマ5、第2のタイマ6、論理回路7）を接続回路4を用いてプログラムで接続することにより、外部入力信号EXT_INのハイ時間及びロー時間を測定する機能を実現することができる。尚、外部入力信号EXT_INのハイ時間及びロー時間を測定するために2つのタイマを使用しているが、これら2つのタイマは1つの割り込み機能付き入力ポートによりコントロールされており、複数の入力ポートを1つの測定に費やすことはない。また、図21を用いて説明した従来技術に比して、同等のマイクロプロセッサ処理であれば、処理が間に合わなくなって次の測定結果がレジスタに上書きされてデータを壊してしまう危険性が低い。これら2点からも解るように、特定用途への専用マイクロコンピュータの周辺回路と同じような無駄のない高機能な周辺回路を構築することが可能である。

【0081】

所定の機能をもった周辺回路を構築するためのプログラムのフローチャートの別の例を図15に示す。まず、レジスタReg0=1、レジスタReg1=1、レジスタReg2=0、レジスタReg3=0、レジスタReg4=0、レジスタReg5=1、レジスタReg6=0、レジスタReg7=0、レジスタRe

g 8 = 1、レジスタ R e g 9 = 0、レジスタ R e g 10 = 0、レジスタ R e g 11 = 0、レジスタ R e g 12 = 0、レジスタ R e g 13 = 0、レジスタ R e g 14 = 1、レジスタ R e g 15 = 0、レジスタ R e g 16 = 0、レジスタ R e g 17 = 1、レジスタ R e g 18 = 0、レジスタ R e g 19 = 0、レジスタ R e g 20 = 0、レジスタ R e g 21 = 1、レジスタ R e g 22 = 0、レジスタ R e g 23 = 0、レジスタ R e g 24 = 0、レジスタ R e g 25 = 0、レジスタ R e g 26 = 1、レジスタ R e g 27 = 0、レジスタ R e g 28 = 0、レジスタ R e g 29 = 1にする (S 4 0 1)。

【0082】

次に、アクセスするアドレスを接続回路4のセクタ401～410に設定する、すなわち、アドレス0xF0000008にライトアクセスする (S 4 0 2)。これにより、セクタ401～410の端子C Sに入力される信号C P U _ W R _ S 1がハイレベルになり、セクタ401～410で選択される信号が切り換えられる。

【0083】

次に、レジスタ R e g 0 = 0、レジスタ R e g 1 = 0、レジスタ R e g 2 = 1、レジスタ R e g 3 = 0、レジスタ R e g 4 = 1、レジスタ R e g 5 = 0にする (S 4 0 3)。次に、アクセスするアドレスを接続回路4のセクタ411及び412に設定する、すなわち、アドレス0xF000000Cにライトアクセスする (S 4 0 4)。これにより、セクタ411及び412の端子C Sに入力される信号C P U _ W R _ S 2がハイレベルになり、セクタ411及び412で選択される信号が切り換えられる。

【0084】

S 4 0 1～S 4 0 4の処理により、論理回路7のインバータ707からの出力信号O U T 2、出力レジスタ3の端子O 1 7、O 1 8からの出力信号O U T R E G 1 7、O U T R E G 1 8が、それぞれ第1のタイマ5の端子S t a r t への入力信号S T A R T 1、端子R e s e t への入力信号R E S E T 1、端子S t o p への入力信号S T O P 1になる。

【0085】

また、第1のタイマ5の端子Overflowからの出力信号OVERFLOW1、出力レジスタ3の端子O20からの出力信号OUTREG20、端子O21からの出力信号OUTREG21が、それぞれ第2のタイマ6の端子Startへの入力信号START2、端子Resetへの入力信号RESET2、端子Stopへの入力信号STOP2になる。

【0086】

また、第1のタイマ5の端子Overflowからの出力信号OVERFLOW1、第2のタイマ6の端子Overflowからの出力信号OVERFLOW2、出力レジスタ3の端子O24からの出力信号OUTREG24、端子O25からの出力信号OUTREG25、端子O26からの出力信号OUTREG26が、それぞれ論理回路7のORゲート701への入力信号IN1、IN2、IN3、ORゲート703及びANDゲート705への入力信号TRG、フリップフロップ702の端子Rへの入力信号D_RESETになる。また、論理回路7のANDゲート706からの出力信号OUT1が外部出力信号EXT_OUTになる。

【0087】

次に、レジスタReg0=0、レジスタReg1=1にする(S405)。次に、アクセスするアドレスを接続回路4のセクタ413~416に設定する、すなわち、アドレス0xF000010にライトアクセスする(S406)。これにより、セクタ413~416の端子CSに入力される信号CPU_WR_S3がハイレベルになり、セクタ413~416で選択される信号が切り換えられる。

【0088】

S405及びS406の処理により、出力レジスタ3の端子O0~O7からの出力信号OUTREG0~OUTREG7が第1のタイマ5の端子IN0~IN7への入力信号DIN1__0~DIN1__7になり、出力レジスタ3の端子O8~O15からの出力信号OUTREG8~OUTREG15が第2のタイマ6の端子IN0~IN7への入力信号DIN2__0~DIN2__7になる。したがって、図16に示すような周辺回路が構築される。

【0089】

次に、レジスタ `Reg 0`～`7` の値を `0x06`、レジスタ `Reg 8`～`15` の値を `0x02`、レジスタ `Reg 17` の値を `1`、レジスタ `Reg 18` の値を `0`、レジスタ `Reg 20` の値を `1`、レジスタ `Reg 21` の値を `0`、レジスタ `Reg 24` の値を `0`、レジスタ `Reg 25` の値を `1`、レジスタ `Reg 26` の値を `1` にする (`S407`)。

【0090】

次に、アクセスするアドレスを出力レジスタ `3` に設定する、すなわち、アドレス `0xF0000004` にライトアクセスする (`S408`)。これにより、出力レジスタ `3` の入力信号 `CPU_WR` がハイレベルになり、出力レジスタ `3` がデータバス `8` 上の信号を取り込む状態になる。

【0091】

`S407` 及び `S408` の処理により、第 `1` のタイマ `5` 及び第 `2` のタイマ `6` では、端子 `IN 0`～`IN 7` への入力値がそれぞれ `0x06`、`0x02` である状態で、端子 `Reset` への入力が高レベルになる。また、論理回路 `7` では、`OR` ゲート `701` への `3` 入力の全てがローレベル、フリップフロップ `702` の端子 `R` への入力が高レベル、インバータ `703` への入力及び `AND` ゲート `705` への `1` 入力が高レベルになる。したがって、第 `1` のタイマ `5`、第 `2` のタイマ `6` はリセットがかかった状態であり、カウント値はそれぞれ `0x06`、`0x02` に設定されている。また、外部出力信号 `EXT_OUT` はハイレベルに固定された状態になる。

【0092】

外部出力信号 `EXT_OUT` として所定のハイ時間及びロー時間をもったパルスの出力を開始するためのプログラムのフローチャートを図 `17` に示す。まず、レジスタ `Reg 17` の値を `0`、レジスタ `Reg 18` の値を `0`、レジスタ `Reg 20` の値を `0`、レジスタ `Reg 21` の値を `0`、レジスタ `Reg 24` の値を `1`、レジスタ `Reg 25` の値を `1`、レジスタ `Reg 26` の値を `0` にする (`S501`)。次に、アクセスするアドレスを出力レジスタ `3` に設定する、すなわち、アドレス `0xF0000004` にライトアクセスする (`S502`)。次に、レジスタ `Reg 24` の値を `0` にする (`S503`)。次に、アクセスするアドレスを出力レジスタ `3` に設定する (`S504`)。

【0093】

この処理により、第1のタイマ5及び第2のタイマ6の端子Resetへの入力がローレベルに反転して、第1のタイマ5及び第2のタイマ6のリセットが解除される。また、論理回路7では、ORゲート701の1入力がハイレベルに反転した後、再びローレベルに反転するとともに、フリップフロップ702の端子Rへの入力がローレベルに反転する。したがって、外部出力信号EXT_OUTがローレベルに反転するとともに、第1のタイマ5の端子Startへの入力がハイレベルに反転して第1のタイマ5がカウント値0x06からカウントを開始する。

【0094】

その後、第1のタイマ5のカウント値にオーバーフローが発生すると、第1のタイマ5の端子Overflowからの出力がハイレベルに反転して、第2のタイマ6の端子Startへの入力、及び、論理回路7のORゲート701の1入力がハイレベルに反転する。これにより、第2のタイマ6がカウント値0x02からカウントを開始するとともに、外部出力信号EXT_OUTがハイレベルに、第1のタイマ5の端子Startへの入力がローレベルにそれぞれ反転する。

【0095】

また、第1のタイマ5のオーバーフローに伴って割り込み要求が発生するが、この割り込み要求が発生すると、図18に示すフローチャートのプログラムが実行される。まず、レジスタReg0～7の値を0x04、レジスタReg17の値を1、レジスタReg18の値を0、レジスタReg20の値を0、レジスタReg21の値を0、レジスタReg24の値を0、レジスタReg25の値を1、レジスタReg26の値を0にする(S601)。次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする(S602)。次に、レジスタReg17の値を0にする(S603)。次に、アクセスするアドレスを出力レジスタ3に設定する(S604)。

【0096】

この処理により、第1のタイマ5では、端子IN0～IN7への入力値が0x04である状態で、端子Resetへの入力がハイレベルになった後、再びローレベルになる。したがって、第1のタイマ5には一時的にリセットがかかり、そのカ

ウント値が0x04に設定されるとともに、その端子Overflowからの出力がローレベルに反転して、論理回路7のORゲート701の3入力全てが全てローレベルになる。

【0097】

その後、第2のタイマ6のカウント値にオーバーフローが発生すると、第2のタイマ6の端子Overflowからの出力がハイレベルに反転して、論理回路7のORゲート701の1入力がハイレベルに反転する。これにより、外部出力信号EXT_OUTがローレベルに反転するとともに、第1のタイマ5の端子Startへの入力がハイレベルに反転して、第1のタイマ5がカウント値0x04からカウントを開始する。

【0098】

また、第2のタイマ6のオーバーフローに伴って割り込み要求が発生するが、この割り込み要求が発生すると、図19に示すフローチャートのプログラムが実行される。まず、レジスタReg8～15の値を0x0A、レジスタReg17の値を0、レジスタReg18の値を0、レジスタReg20の値を1、レジスタReg21の値を0、レジスタ24の値を0、レジスタReg25の値を1、レジスタReg26の値を0にする（S701）。次に、アクセスするアドレスを出力レジスタ3に設定する、すなわち、アドレス0xF0000004にライトアクセスする（S702）。次に、レジスタReg20の値を0にする（S703）。次に、アクセスするアドレスを出力レジスタ3に設定する（S704）。

【0099】

この処理により、第2のタイマ6では、端子IN0～IN7への入力値が0x0Aである状態で、端子Resetへの入力がハイレベルになった後、再びローレベルになる。したがって、第2のタイマ6には一時的にリセットがかかり、そのカウント値が0x0Aに設定されるとともに、その端子Overflowからの出力がローレベルに反転して、論理回路7のORゲート701の3入力全てが全てローレベルになる。

【0100】

以上のプログラムにより、第1のタイマ5のカウント値T1、第2のタイマ6

のカウンタ値T2、レジスタReg0～7の値、レジスタReg8～15の値、及び、外部出力信号EXT__OUTが図20に示すように変化する。尚、第1のタイマ5及び第2のタイマ6のクロック信号の周波数を4 [MHz]である場合を想定している。

【0101】

まず、図15に示したプログラムが実行されただけの初期状態では、外部出力信号EXT__OUTがハイレベルに固定されている。また、第1のタイマ5、第2のタイマ6にはリセットがかかっており、それらのカウンタ値はそれぞれ0x06、0x02に設定されている。図17に示したプログラムが実行されると、図20中にAで示すように、外部出力信号EXT__OUTがローレベルに反転するとともに、第1のタイマ5がカウンタ値0x06からカウンタを開始する。

【0102】

その後、1.5 [μ s]が経過すると、第1のタイマ5にオーバーフローが発生するので、図18に示したプログラムが実行され、図20中にBで示すように、外部出力信号EXT__OUTがハイレベルに反転する。また、第2のタイマ6がカウンタ値0x02からカウンタを開始する。また、第1のタイマ5にリセットがかかり、そのカウンタ値が0x04に設定される。

【0103】

その後、0.5 [μ s]が経過すると、第2のタイマ6にオーバーフローが発生するので、図19に示したプログラムが実行され、図20中にCで示すように、外部出力信号EXT__OUTがローレベルに反転する。また、第1のタイマ6がカウンタ値0x04からカウンタを開始する。また、第2のタイマ6にリセットがかかり、そのカウンタ値が0x0Aに設定される。

【0104】

その後、1.0 [μ s]が経過すると、第1のタイマ5にオーバーフローが発生するので、図18に示したプログラムが実行され、図20中にDで示すように、外部出力信号EXT__OUTがハイレベルに反転する。また、第2のタイマ6がカウンタ値0x0Aからカウンタを開始する。また、第1のタイマ5にリセットがかかり、そのカウンタ値が0x04に設定される。

【0105】

その後、2.5 [μ s]が経過すると、第2のタイマ6にオーバーフローが発生するので、図19に示したプログラムが実行され、図20中にEで示すように、外部出力信号EXT__OUTがローレベルに反転する。また、第1のタイマ6がカウント値0x04からカウントを開始する。また、第2のタイマ6にリセットがかかり、そのカウント値が0x0Aに設定される。

【0106】

このように、本実施形態のマイクロコンピュータでは、個々の基本的な周辺回路（第1のタイマ5、第2のタイマ6、論理回路7）を接続回路4を用いてプログラムで接続することにより、外部出力信号EXT__OUTとして所望のハイ時間及びロー時間をもったパルスを生成することができる。

【0107】

尚、本実施形態では、基本的な周辺回路として第1のタイマ5、第2のタイマ6、及び、論理回路7を設けていたが、他の汎用的な周辺回路で置き換えた形態であったり、他の周辺回路を追加した形態であってもよい。また、周辺回路に対してデータをリードライトするための回路の数、ビット幅、アドレスマッピングなども他の形態であっても構わない。また、接続回路で周辺回路への入出力を選択する方法についても、組み合わせる入出力端子の比率、数、アドレスマッピング、設定ビットなども他の形態であっても構わない。

【0108】

【発明の効果】

以上説明したように、本発明のマイクロコンピュータでは、個々の基本的な周辺回路を、接続のための回路を用いてプログラムで接続することによって、個々の基本的な周辺回路では実現し得ない高機能を実現するようになっている。したがって、汎用のマイクロコンピュータを使用する場合のように過剰な機能をもった機種を選択する羽目になってコスト高を招くという事態を防止することができる。また、必要になることがわかっている基本的な周辺回路のみを組み込んでいれば、周辺回路の詳細な設計は不要になるので、マイクロコンピュータを組み込む装置全体の開発期間を短縮することができる。さらに、専用の周辺回路を備え

たマイクロコンピュータの場合、ソフトウェア開発用のブレッドボードを用意する必要があるが、本発明のマイクロコンピュータでは、ソフトウェアのデバッグ時に周辺回路を決定でき、このことも開発期間の短縮に貢献する。

【0109】

まとめると、本発明のマイクロコンピュータによれば、特殊な製造技術や製造工程を使用することなく、また、専用マイクロコンピュータのように長期の開発日程を要することなく、汎用マイクロコンピュータよりも専用マイクロコンピュータに近い高機能な周辺回路機能を実現することができる。さらに、ソフトウェアの設計時に、より効率の良い周辺回路に変更することが可能である。

【図面の簡単な説明】

【図1】 本発明の一実施形態であるマイクロコンピュータのブロック図である。

【図2】 第1のタイマを構成するダウンカウンタを示す図である。

【図3】 第2のタイマを構成するダウンカウンタを示す図である。

【図4】 論理回路の構成を示す図である。

【図5】 入力レジスタを構成するラッチ回路を示す図である。

【図6】 出力レジスタを構成するラッチ回路を示す図である。

【図7】 接続回路を構成するセレクタを示す図である。

【図8】 接続回路を構成する他のセレクタを示す図である。

【図9】 図7に示したセレクタにおいて、端子に入力される信号の状態と選択される端子との関係を示す図である。

【図10】 所定の機能をもった周辺回路を構築するためのプログラムのフローチャートである。

【図11】 図10に示したプログラムで構築される周辺回路の構成を示す図である。

【図12】 図11に示した構成の周辺回路を制御するためのプログラムのフローチャートである。

【図13】 図11に示した構成の周辺回路を制御するための別のプログラムのフローチャートである。

【図 14】 図 11 に示した構成の周辺回路による動作を説明するための図である。

【図 15】 所定の機能をもった周辺回路を構築するための別のプログラムのフローチャートである。

【図 16】 図 15 に示したプログラムで構築される周辺回路の構成を示す図である。

【図 17】 図 16 に示した構成の周辺回路を制御するためのプログラムのフローチャートである。

【図 18】 図 16 に示した構成の周辺回路を制御するための別のプログラムのフローチャートである。

【図 19】 図 16 に示した構成の周辺回路を制御するためのさらに別のプログラムのフローチャートである。

【図 20】 図 16 に示した構成の周辺回路による動作を説明するための図である。

【図 21】 従来技術における外部入力信号のハイ時間及びロー時間の測定動作を説明するための図である。

【図 22】 図 21 に示した動作において必要となるプログラムのフローチャートである。

【図 23】 従来技術における外部入力信号のハイ時間及びロー時間の別の測定動作を説明するための図である。

【図 24】 図 23 に示した動作において必要となるプログラムのフローチャートである。

【図 25】 図 23 に示した動作において必要となる別のプログラムのフローチャートである。

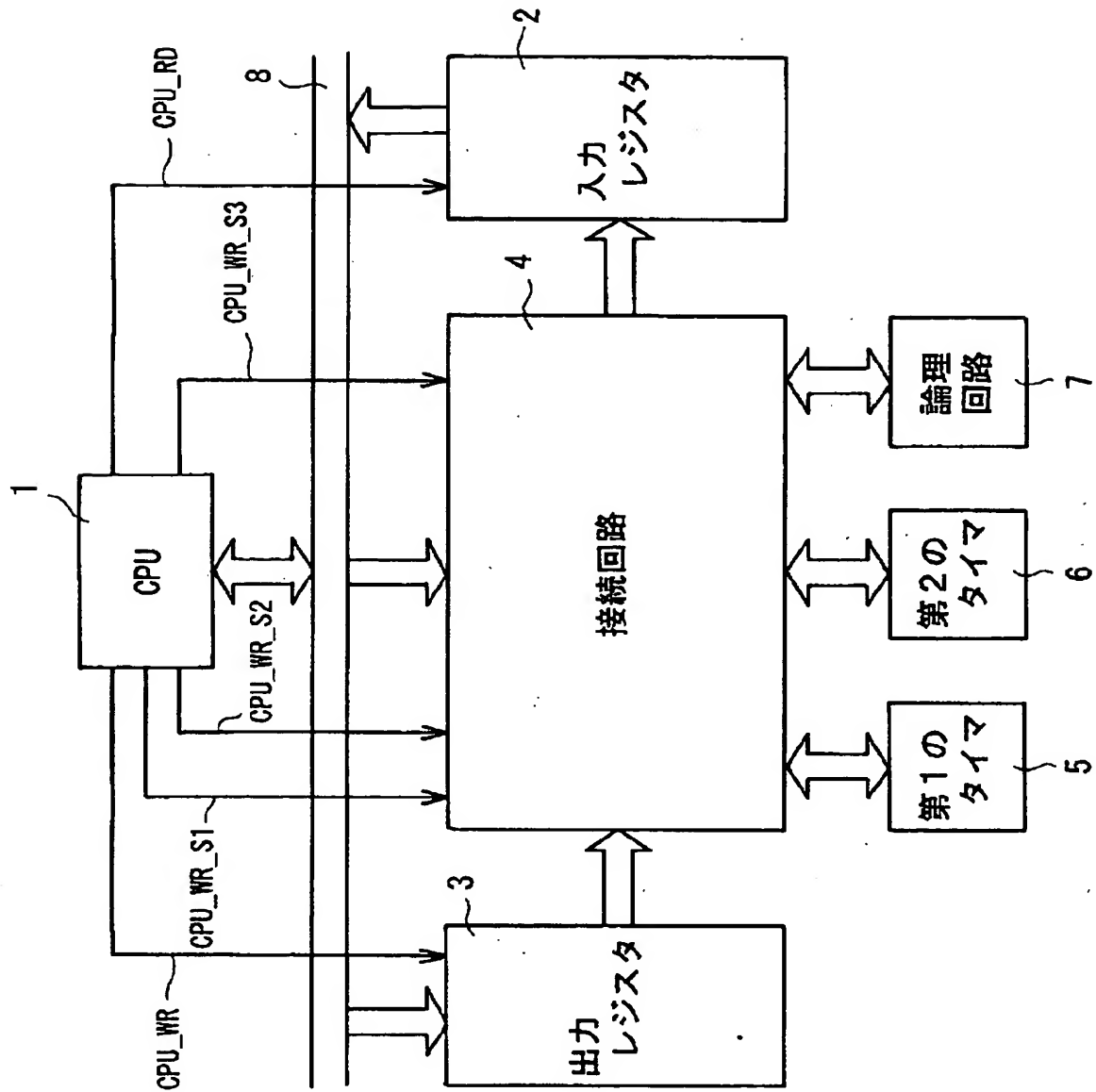
【符号の説明】

- 1 CPU
- 2 入力レジスタ
- 3 出力レジスタ
- 4 接続回路

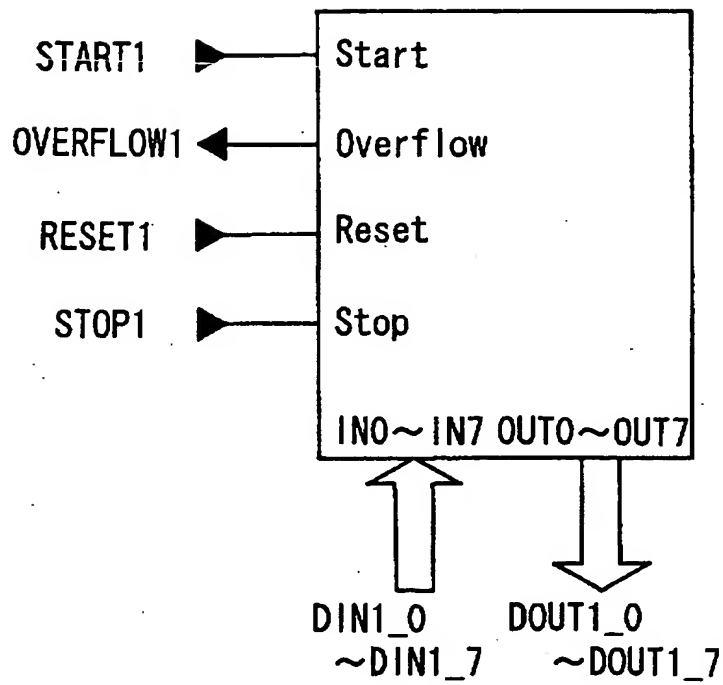
- 5 第1のタイマ
- 6 第2のタイマ
- 7 論理回路
- 8 データバス
- 401~412 セレクタ
- 413~416 セレクタ
- 701 ORゲート
- 702 フリップフロップ
- 703 インバータ
- 704 ANDゲート
- 705 ANDゲート
- 706 ORゲート
- 707 インバータ

【書類名】 図面

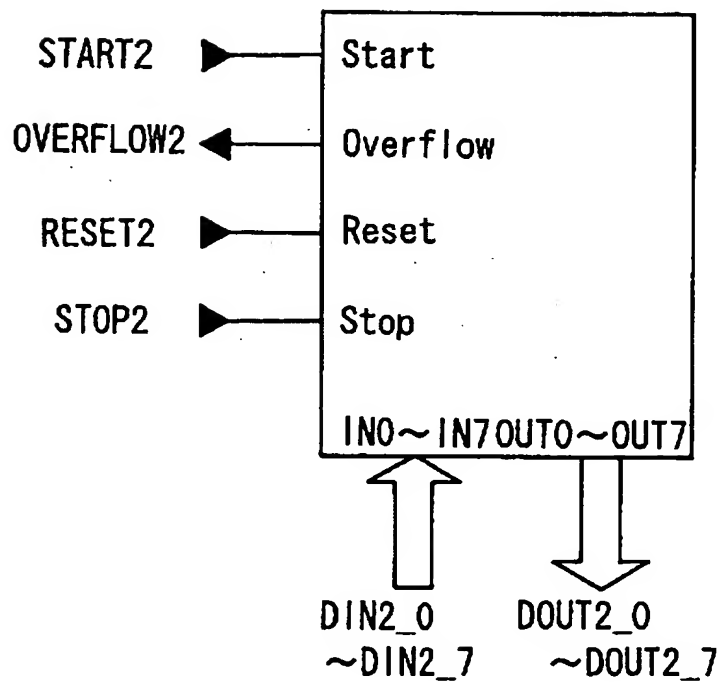
【図 1】



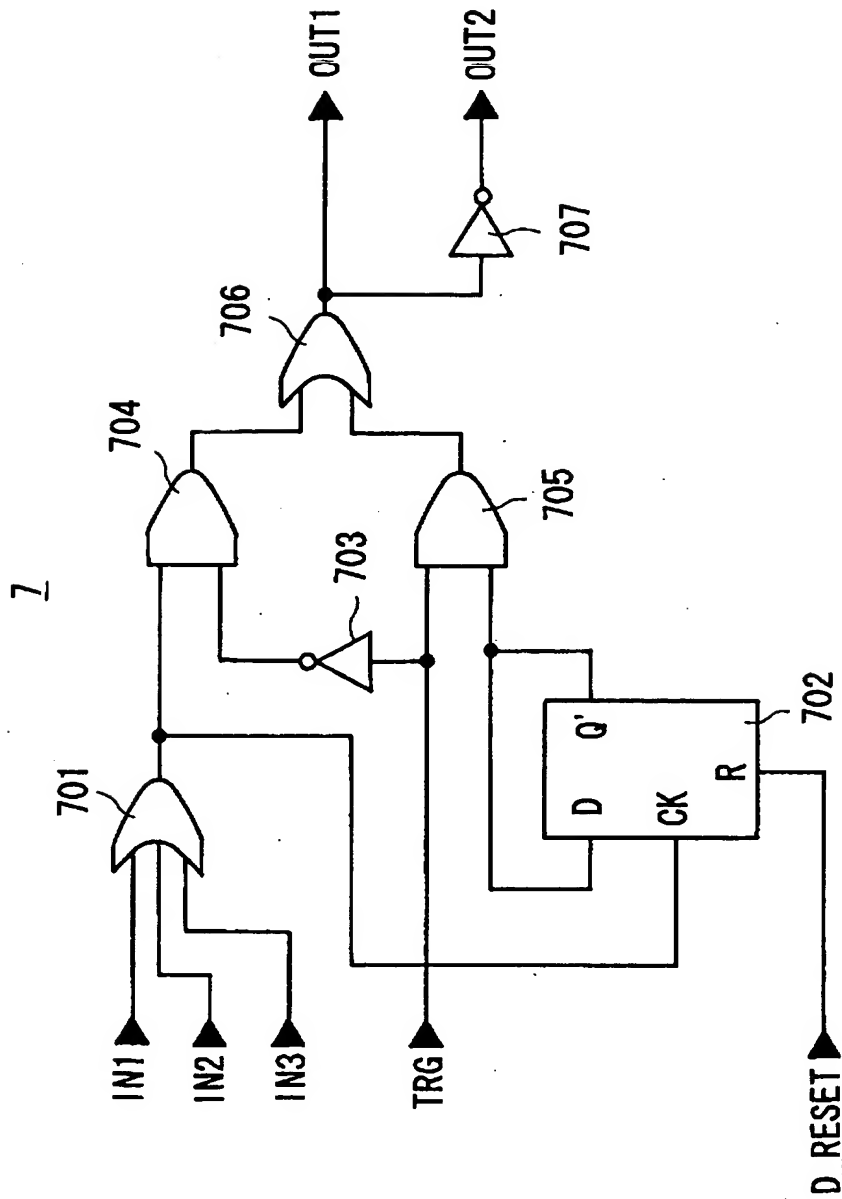
【図 2】



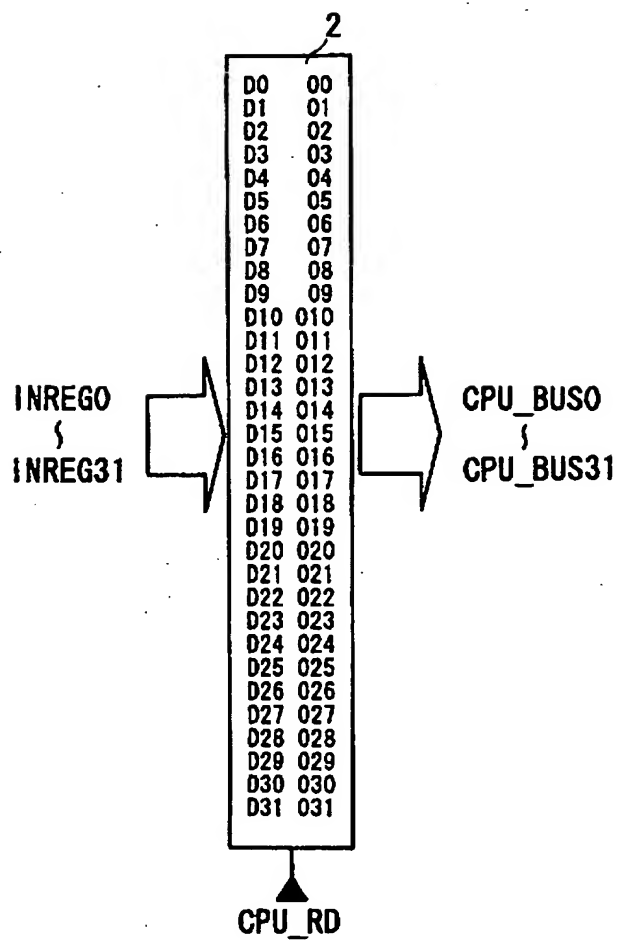
【図 3】



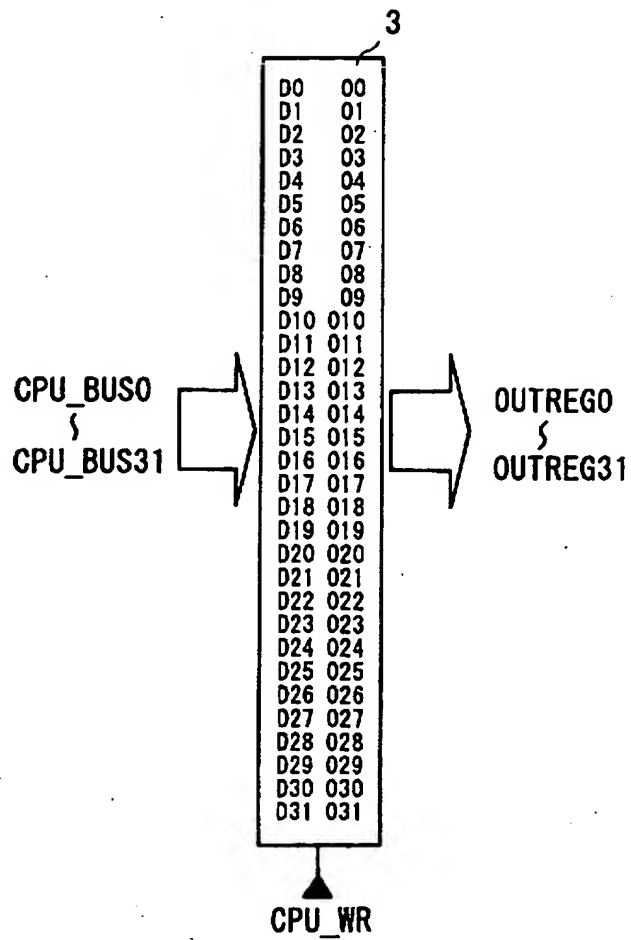
【図 4】



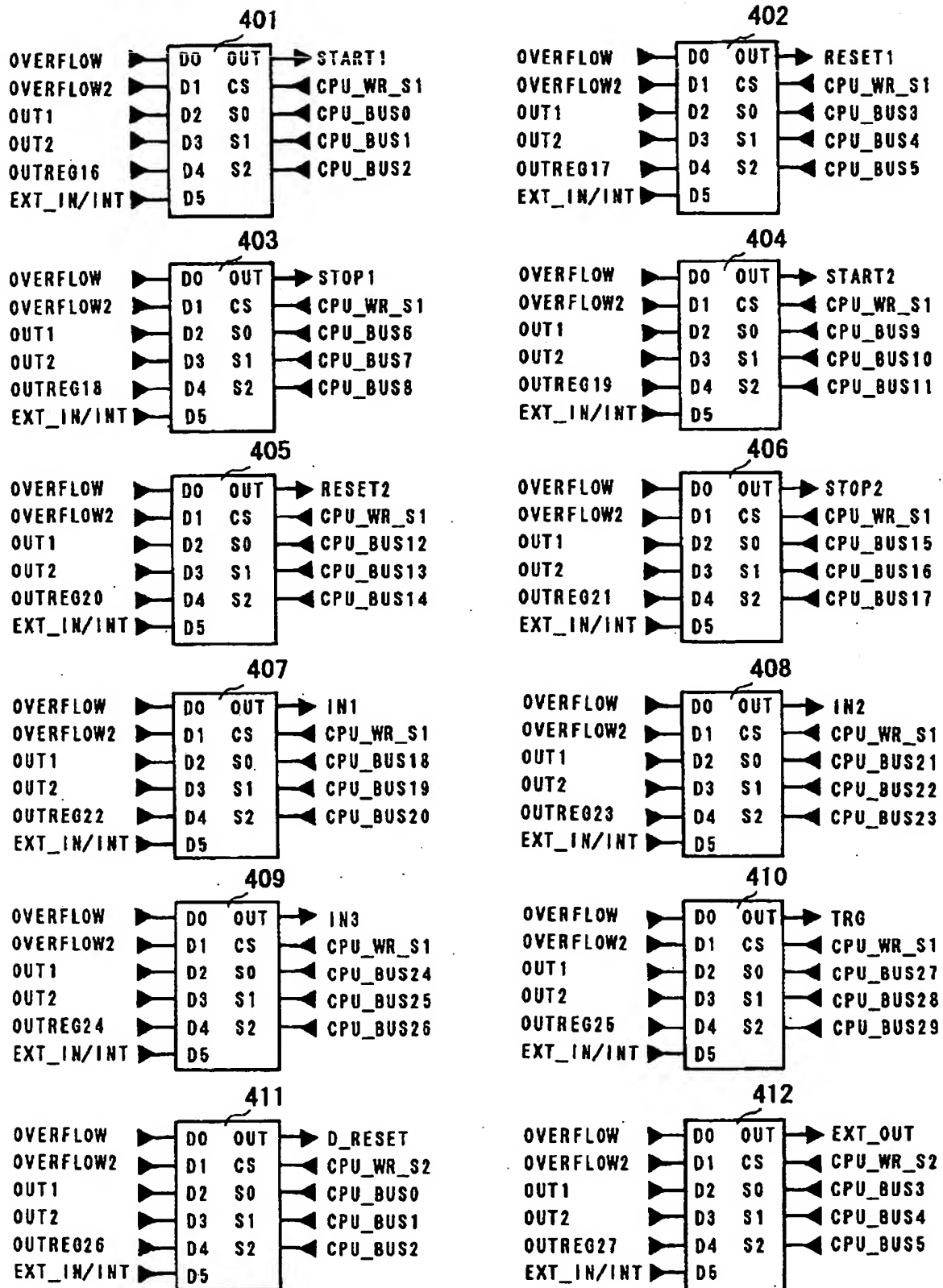
【図 5】



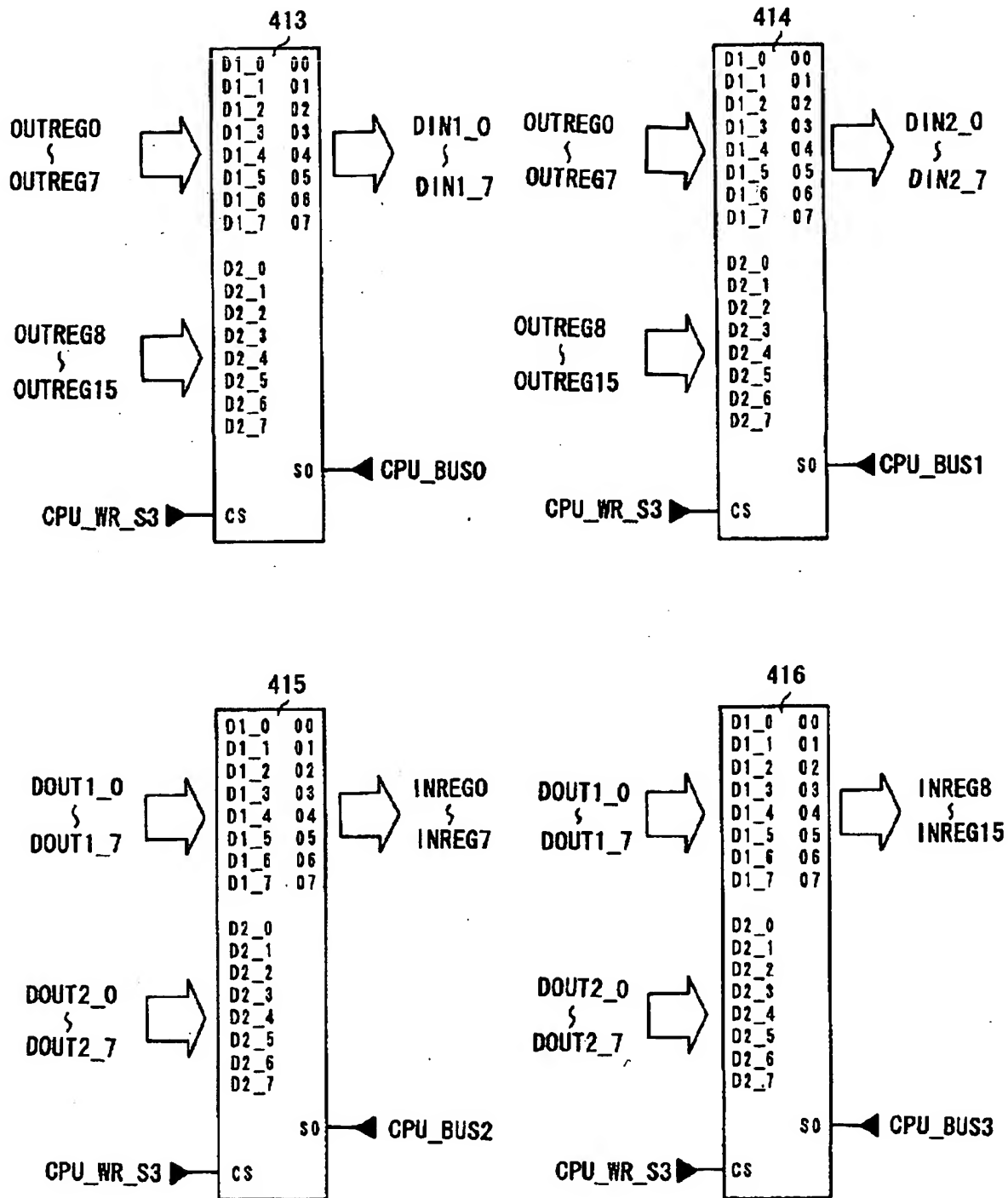
【図 6】



【図 7】



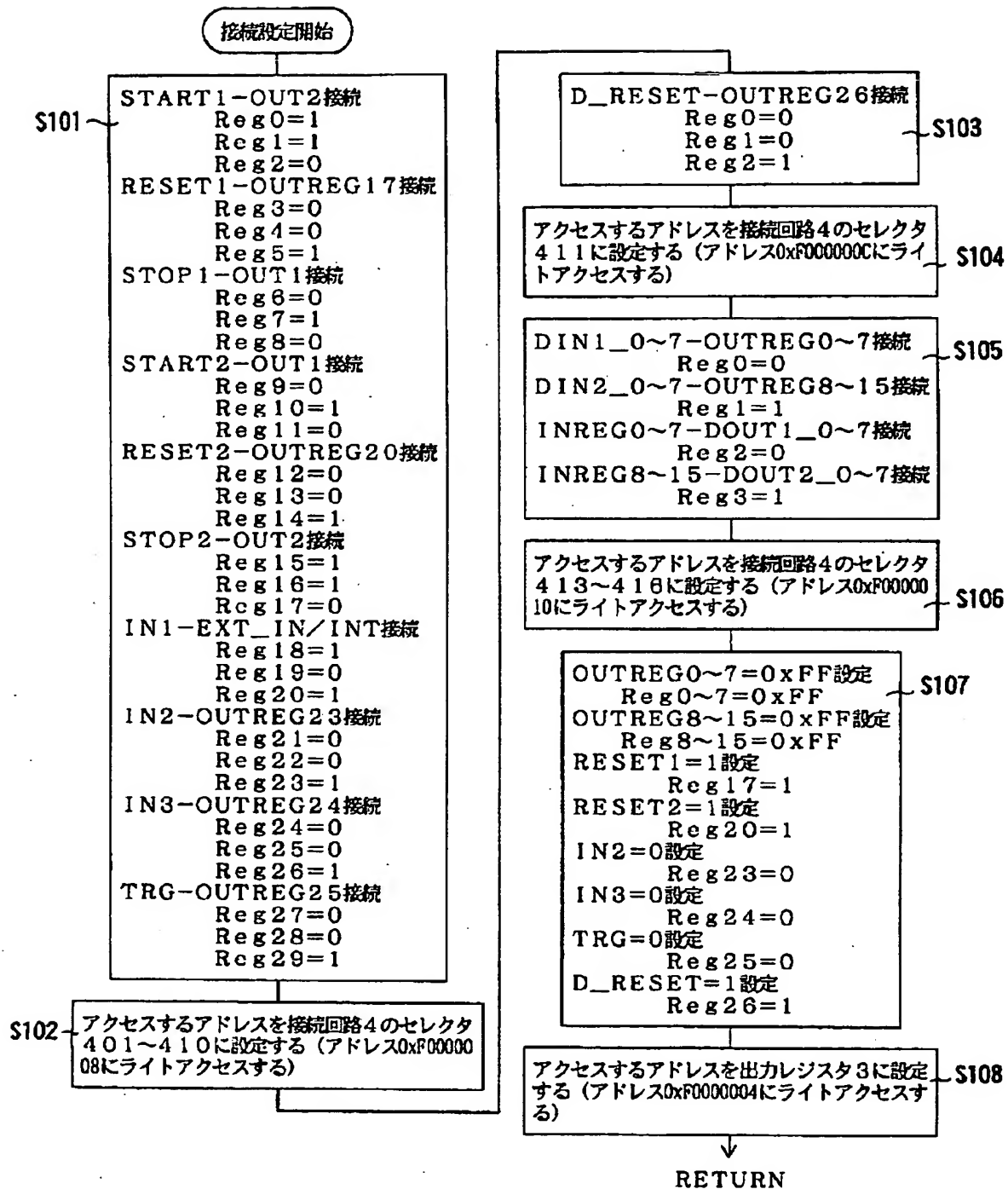
【図 8】



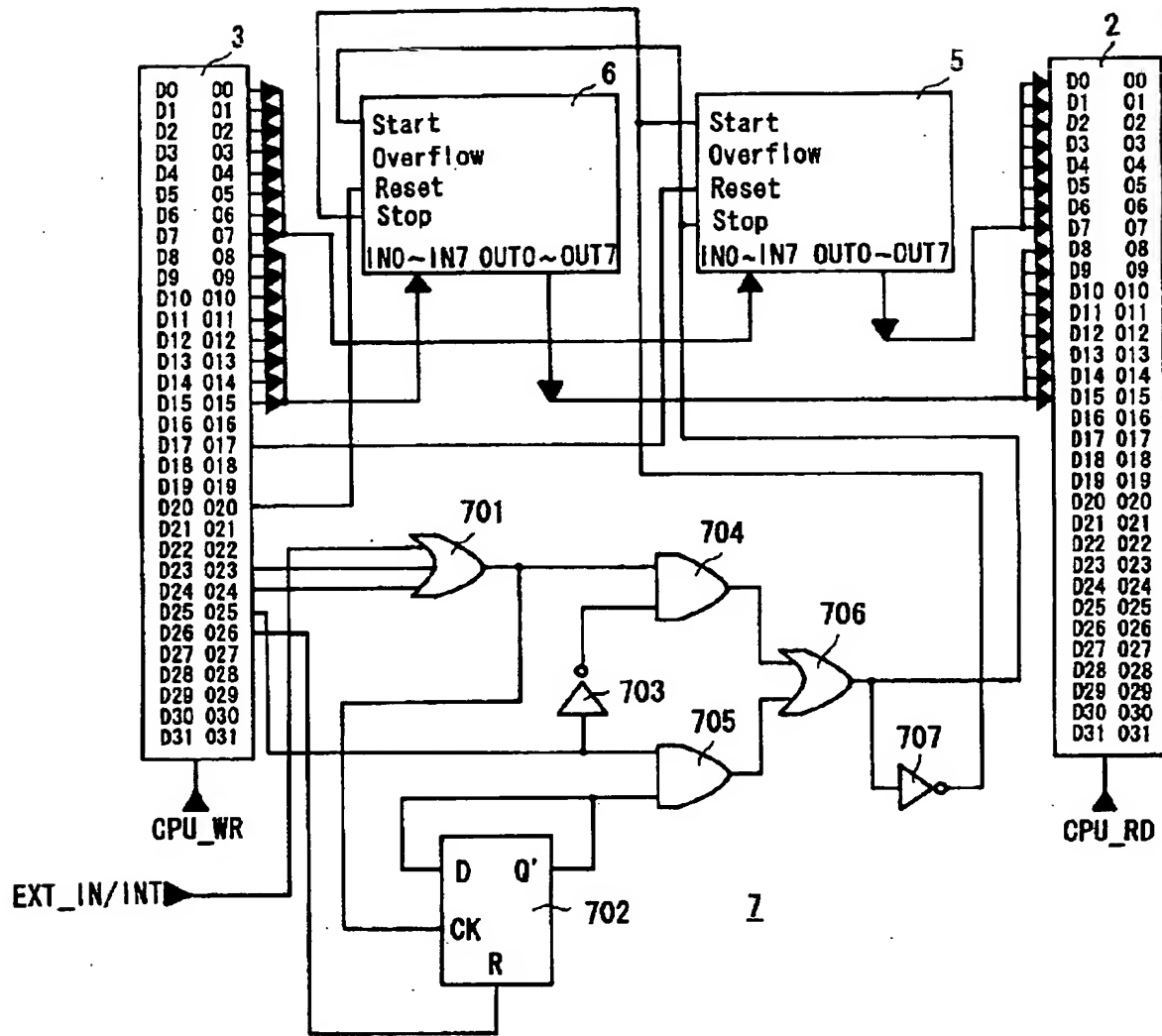
【図 9】

S2	S1	S0	OUT
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	0	0	D4
1	0	1	D5
1	1	0	—
1	1	1	—

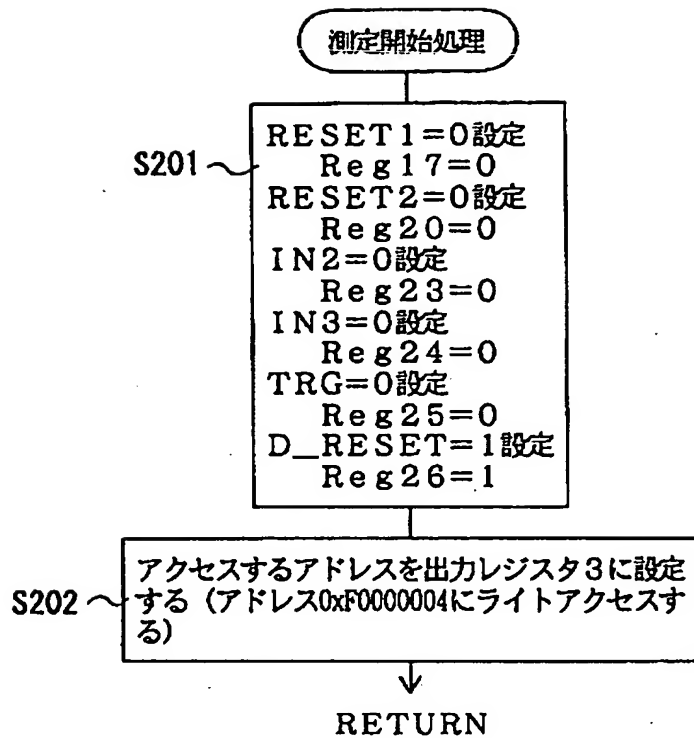
【図10】



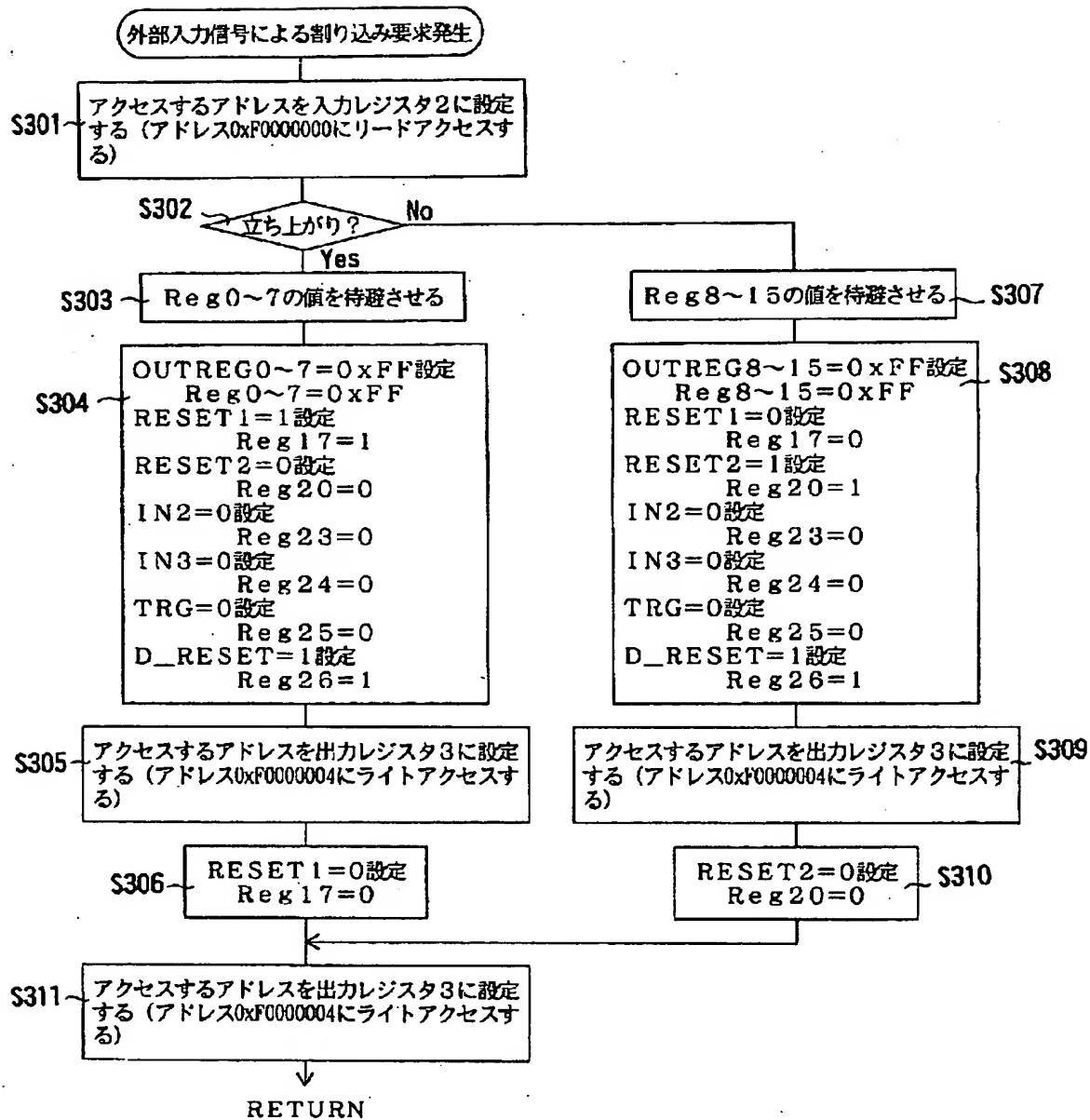
【図 11】



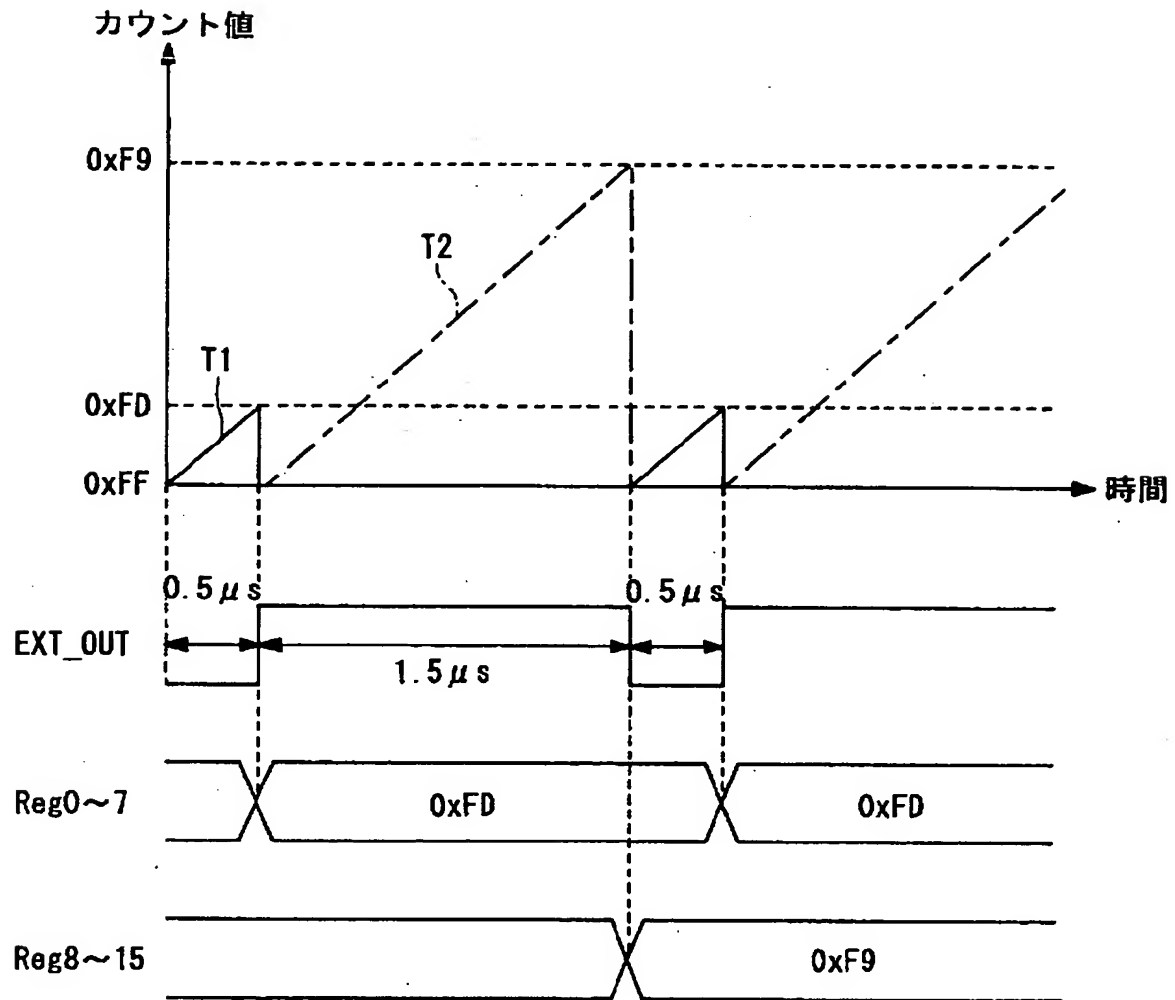
【図 12】



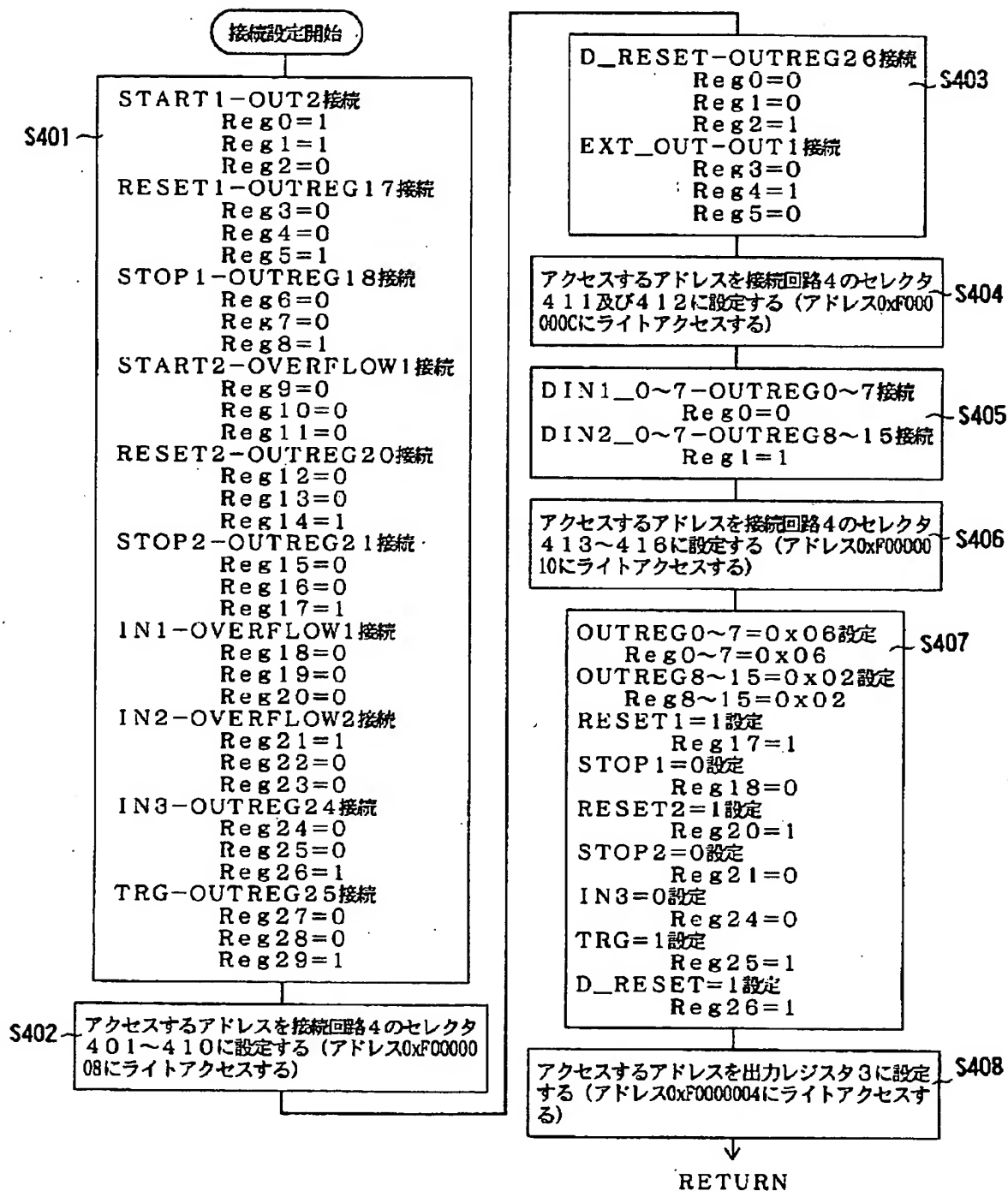
【図 13】



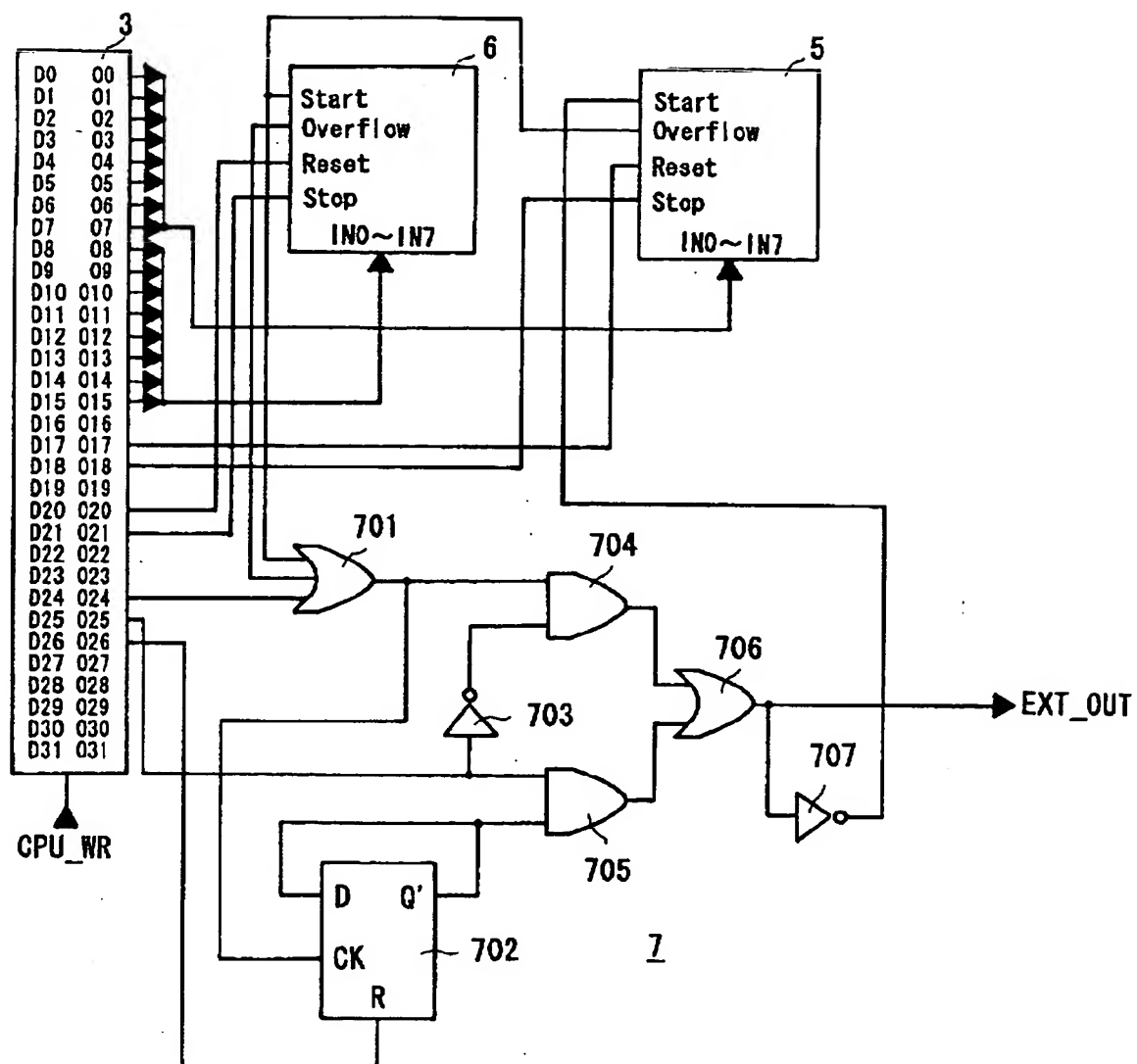
【図 14】



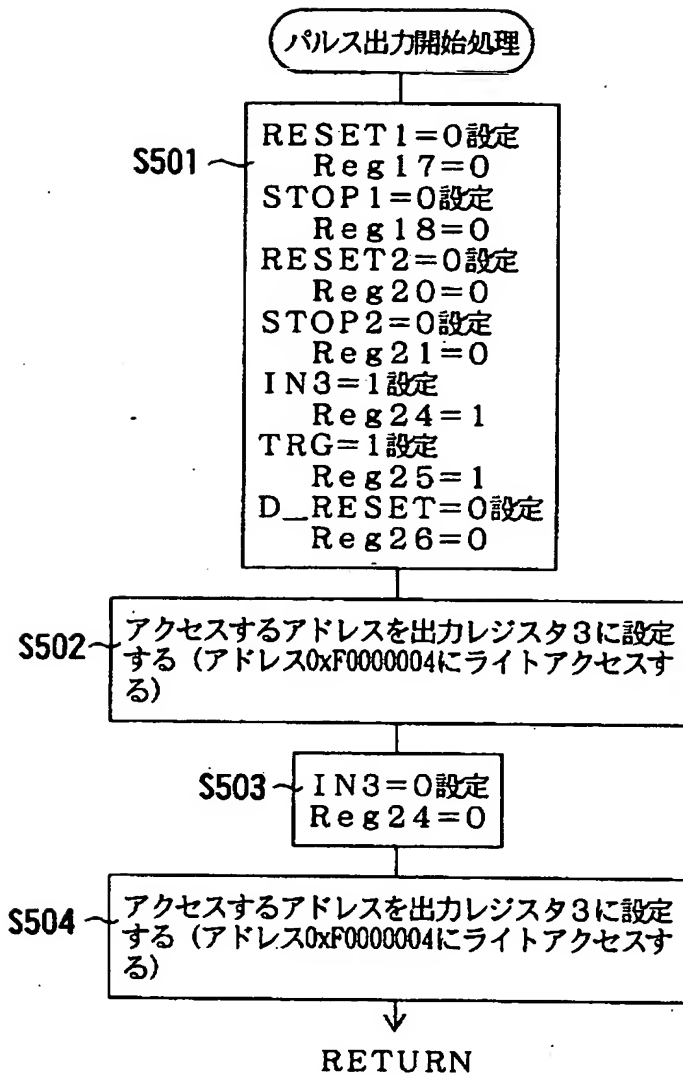
【図 15】



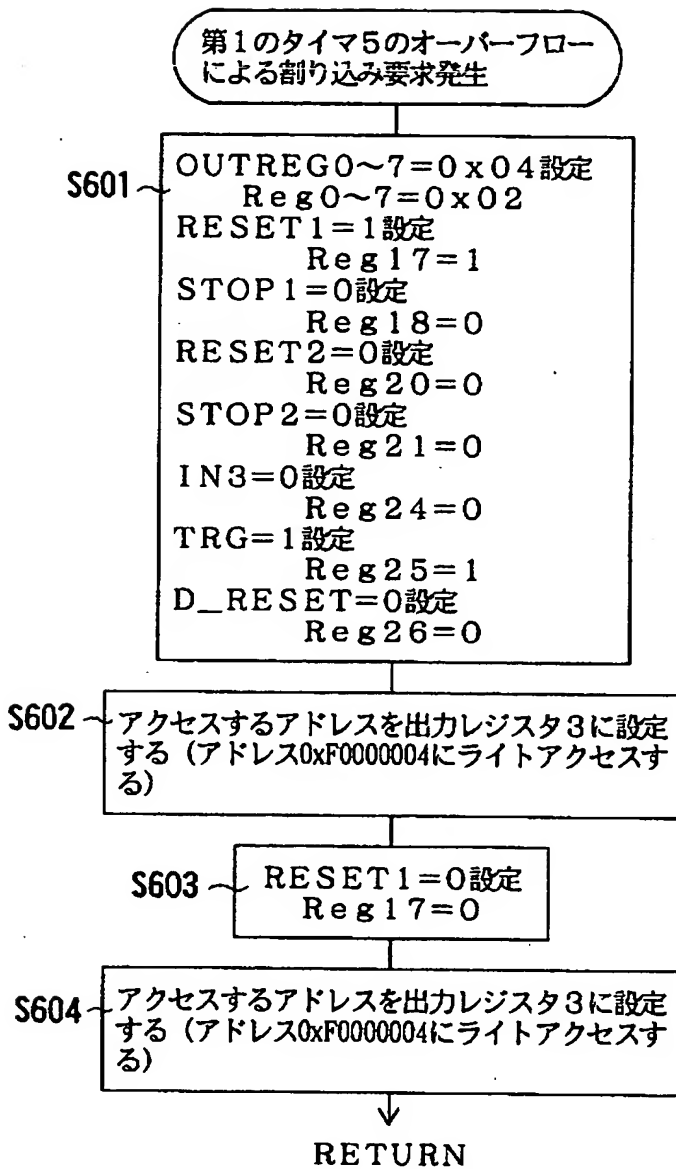
【図 16】



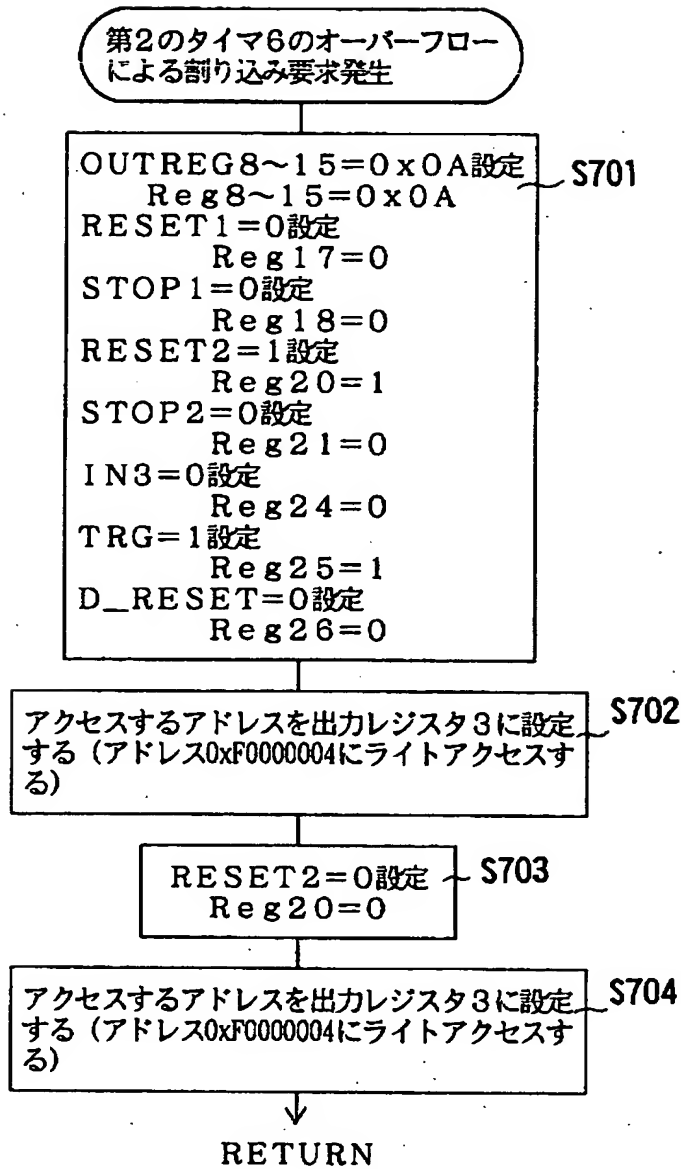
【図 17】



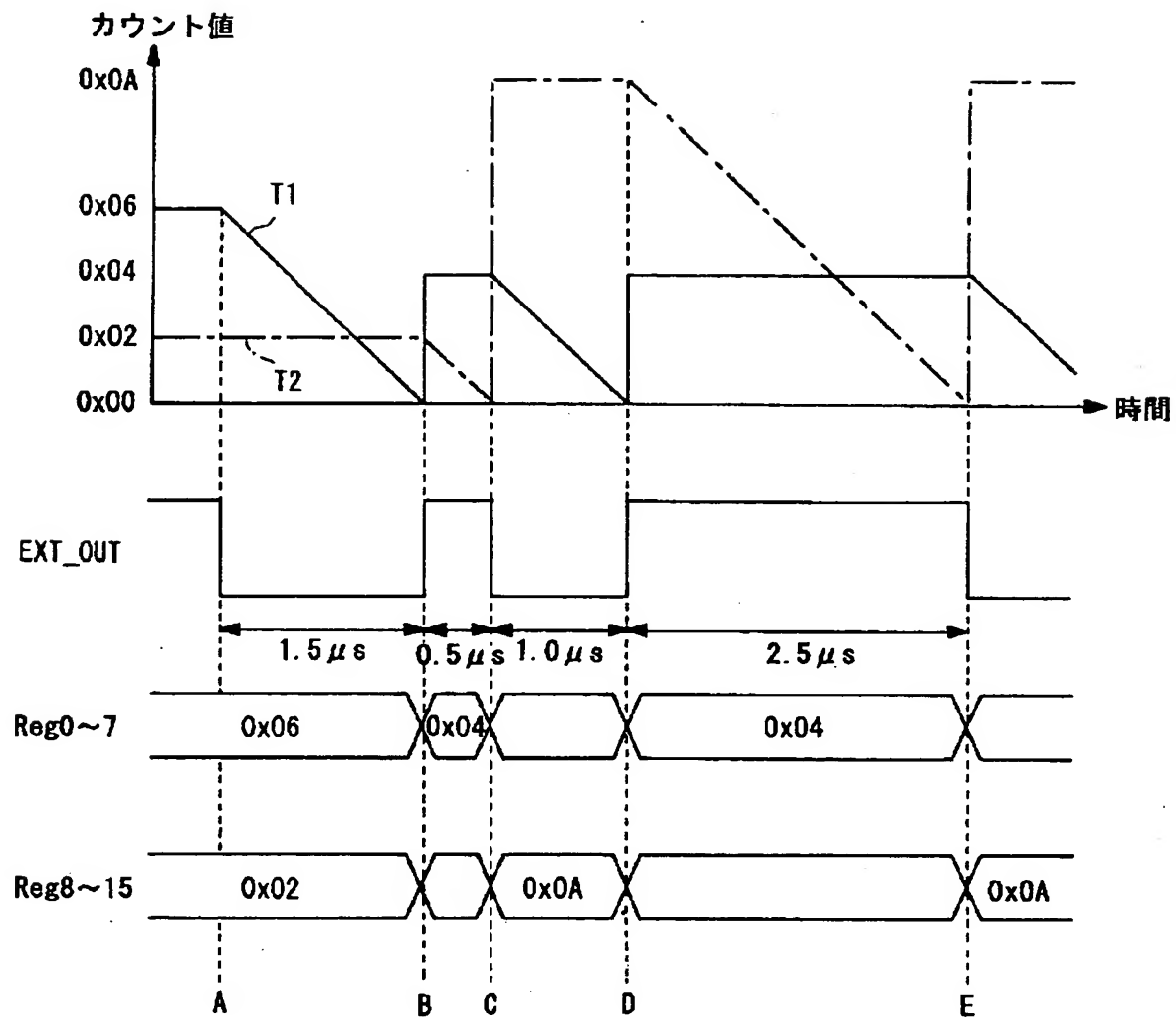
【図 18】



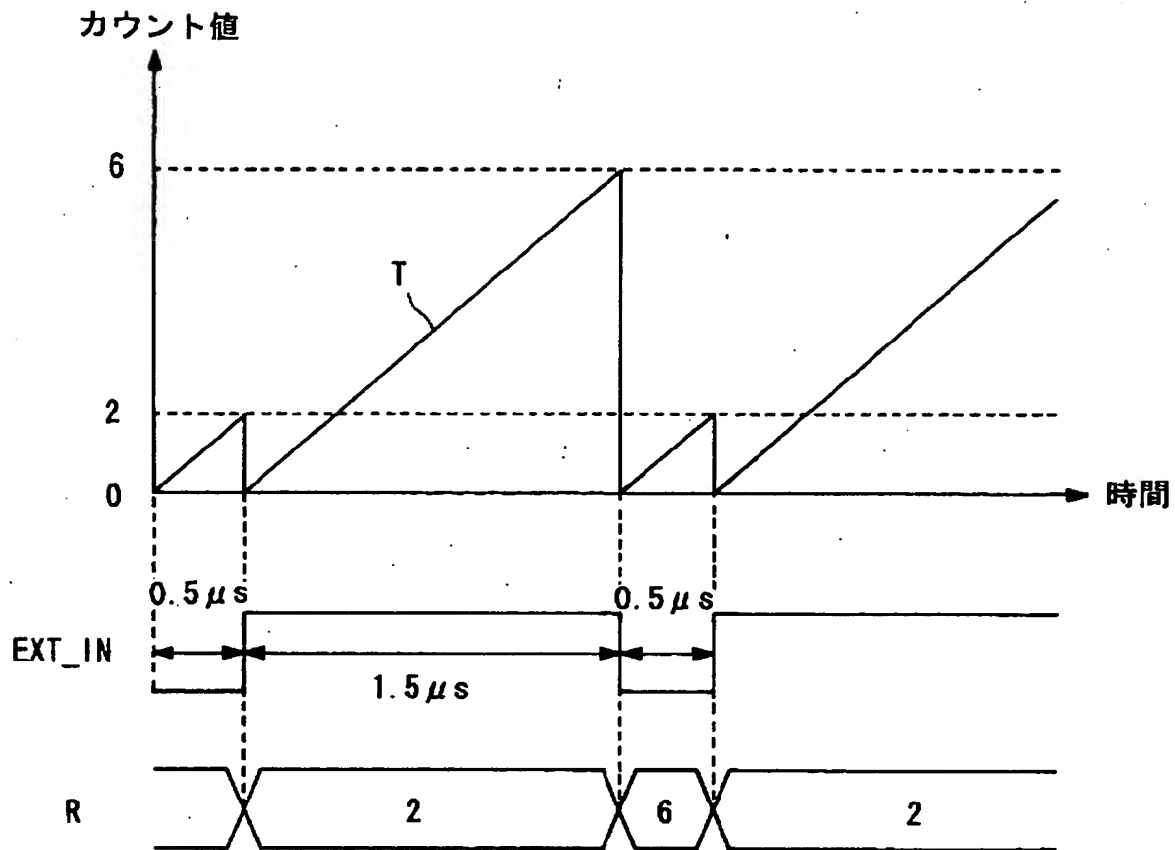
【図19】



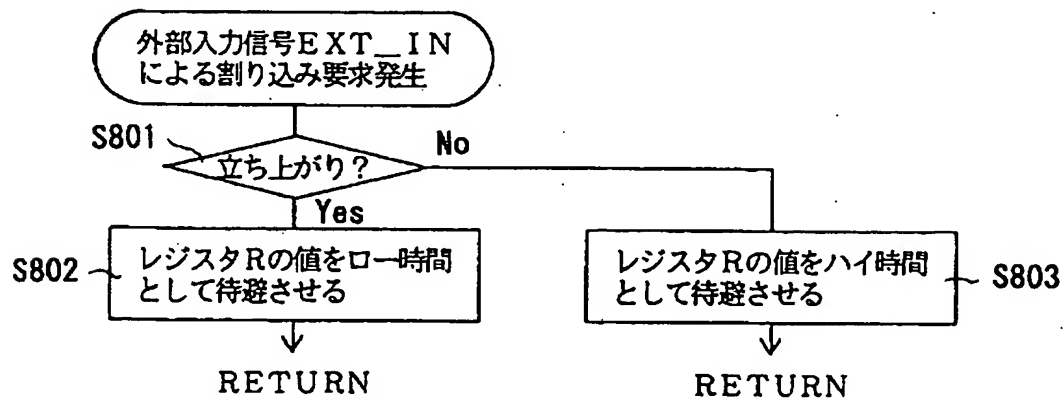
【図 20】



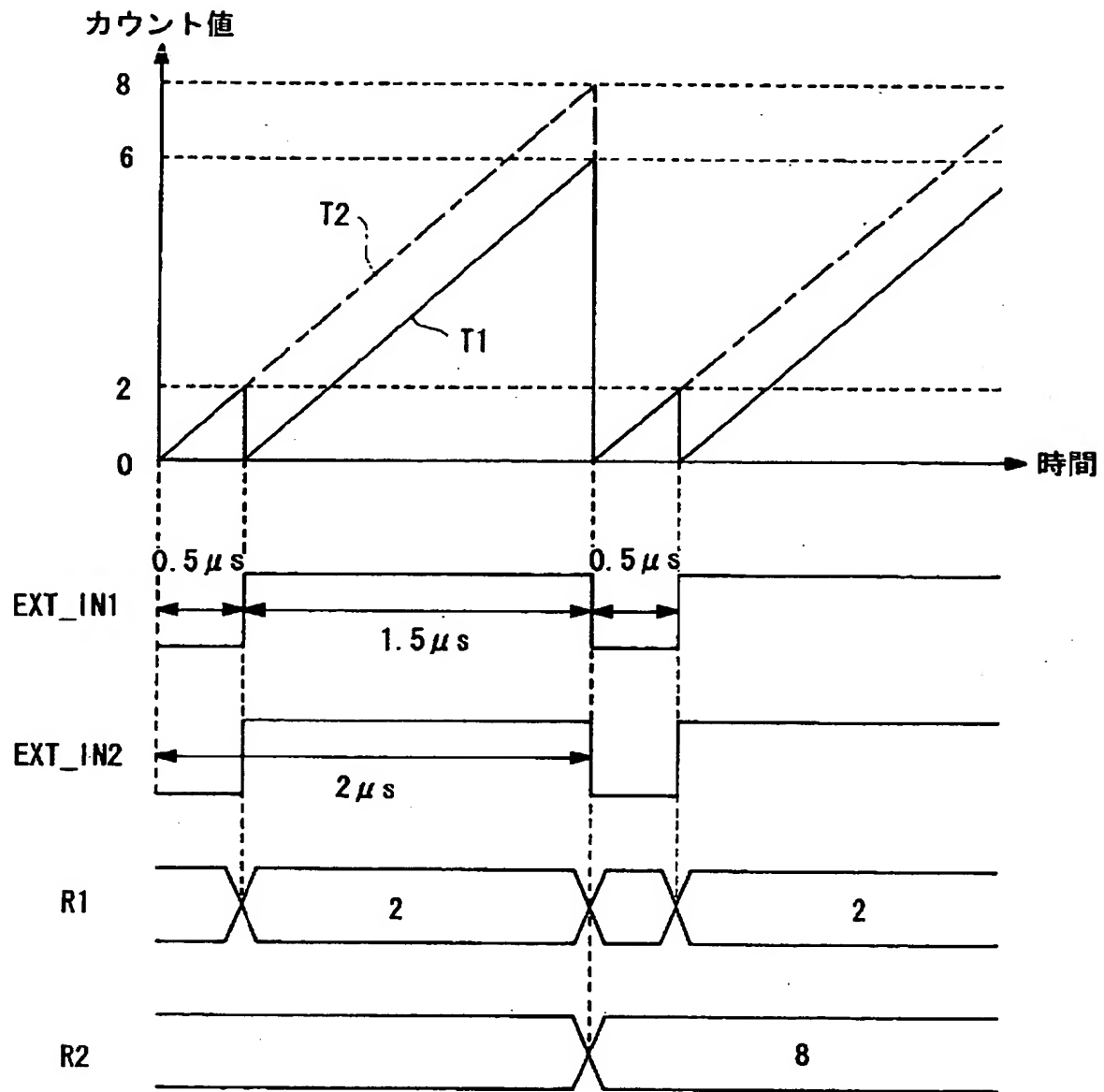
【図 2 1】



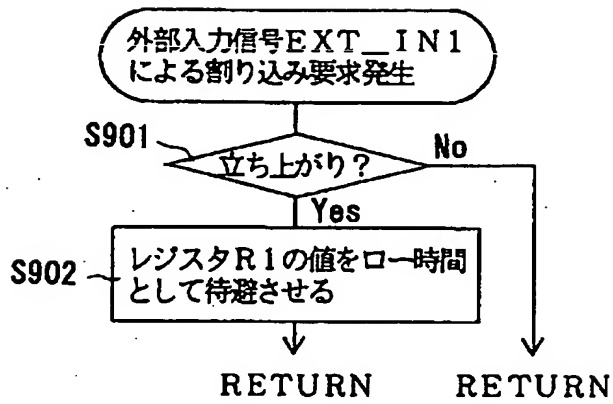
【図 2 2】



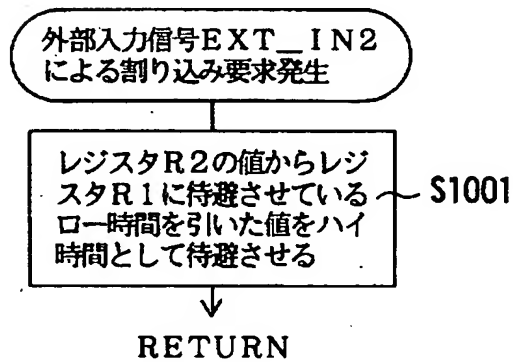
【図 23】



【図 2 4】



【図 2 5】



【書類名】 要約書

【要約】

【課題】 特殊な製造技術や製造工程を使用することなく、また、専用マイクロコンピュータのように長期の開発日程を要することなく、汎用マイクロコンピュータよりも専用マイクロコンピュータに近い高機能な周辺回路機能をソフトウェア処理により実現することができるようにしたマイクロコンピュータを提供する。

【解決手段】 複数の周辺回路 2、3、5、6、7 を備えたマイクロコンピュータにおいて、複数の周辺回路 2、3、5、6、7 間の接続関係をプログラムで制御するための接続回路 4 を備える。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000005049]

1. 変更年月日	1990年 8月29日
[変更理由]	新規登録
住 所	大阪府大阪市阿倍野区长池町22番22号
氏 名	シャープ株式会社